



**TRANSLATION AND ROTATION INVARIANT
MULTISCALE IMAGE REGISTRATION**

THESIS

Jennifer L. Manfra, Second Lieutenant, USAF

AFIT/GE/ENG/02M-16

**DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY**

**AIR FORCE INSTITUTE OF
TECHNOLOGY**

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

Report Documentation Page

Report Date 15 Mar 02	Report Type Final	Dates Covered (from... to) Jun 2001 - Mar 2002
Title and Subtitle Translation and Rotation Invariant Multiscale Image Registration	Contract Number	
	Grant Number	
	Program Element Number	
Author(s) 2ndLt Jennifer L. Manfra, USAF	Project Number	
	Task Number	
	Work Unit Number	
Performing Organization Name(s) and Address(es) Air Force Institute of Technology Graduate School of Engineering and Management, (AFIT/EN) 2950 P Street, Bldg 640 WPAFB, OH 45433-7765	Performing Organization Report Number AFIT/GE/ENG/02M-16	
Sponsoring/Monitoring Agency Name(s) and Address(es) AFRL/IFED ATTN: Maj John G. Keller 32 Brooks Road Rome NY 13441-4114	Sponsor/Monitor's Acronym(s)	
	Sponsor/Monitor's Report Number(s)	
Distribution/Availability Statement Approved for public release, distribution unlimited		
Supplementary Notes		
Abstract The most recent research involved registering images in the presence of translations and rotations using one iteration of the redundant discrete wavelet transform. We extend this work by creating a new multiscale transform to register two images with translation or rotation differences, independent of scale differences between the images. Our two-dimensional multiscale transform uses an innovative combination of lowpass filtering and the continuous wavelet transform to mimic the two-dimensional redundant discrete wavelet transform. This allows us to obtain multiple subbands at various scales while maintaining the desirable properties of the redundant discrete wavelet transform. Whereas the discrete wavelet transform produces results only at dyadic scales, our new multiscale transform produces data at all integer scales. This added flexibility improves registration accuracy without greatly increasing computational complexity and permits accurate registration even in the presence of scale differences.		
Subject Terms Image Registration, Wavelet Transforms, Translation Invariance, Rotation Invariance		

Report Classification unclassified	Classification of this page unclassified
Classification of Abstract unclassified	Limitation of Abstract UU
Number of Pages 113	

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

AFIT/GE/ENG/02M-16

TRANSLATION AND ROTATION INVARIANT
MULTISCALE IMAGE REGISTRATION

THESIS

Presented to the Faculty
Department of Electrical and Computer Engineering
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
in Partial Fulfillment of the Requirements for the
Degree of Master of Science in Electrical Engineering

Jennifer L. Manfra, B.S.E.E. University of Memphis
Second Lieutenant, USAF

March, 2002


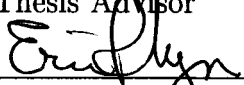
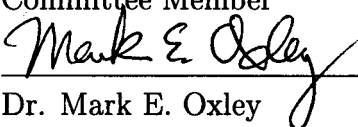
Approved for public release; distribution unlimited

TRANSLATION AND ROTATION INVARIANT
MULTISCALE IMAGE REGISTRATION

Jennifer L. Manfra, B.S.E.E. University of Memphis

Second Lieutenant, USAF

Approved:

 Maj Roger L. Claypoole Jr. Ph.D. Thesis Advisor	<u>13 MAR 02</u> Date
 Maj Eric P. Magee Ph.D. Committee Member	<u>13 MAR 02</u> Date
 Dr. Mark E. Oxley Committee Member	<u>13 March 2002</u> Date

Acknowledgements

I would like to thank my friends and colleagues that have made my time here at AFIT enjoyable. I would like to thank Major Claypoole, Major Magee, and Dr. Mark Oxley for helping with my thesis. I would specifically like to thank Major Claypoole for putting up with me being in his office nonstop for the last year. Thanks for always having time to help and answer any questions that I had about school or the Air Force in general.

I would like to give a special thanks to Ryan, Matt, Russ, Ever, Sebrina, Jake, and Josh who helped me adjust to this new place and this new school. Without all of you, my life would have been miserable. Thank you Ryan for always being there when I needed you.

Thank you to my family for always calling me to give me a break from school work when I was so involved I didn't even realize what time it was.

Jennifer L. Manfra

Table of Contents

	Page
Acknowledgements	iii
List of Figures	vii
Abstract	x
 I. Introduction	 1-1
1.1 Problem Statement	1-1
1.2 Scope	1-2
1.3 Thesis Organization	1-2
 II. Background	 2-1
2.1 Introduction	2-1
2.2 Image Registration Overview	2-1
2.3 Elements of Image Registration	2-2
2.4 Image Registration Techniques	2-3
2.4.1 Fourier-Based Techniques	2-3
2.4.2 Wavelet-Based Techniques	2-4
2.5 The Wavelet Transform	2-6
2.5.1 The Continuous Wavelet Transform	2-7
2.5.2 Function Space Development of the Discrete Wavelet Transform	2-8
2.5.3 The Two-Dimensional Wavelet Transform	2-11
2.5.4 Lowpass Filters	2-17
2.5.5 The Wavelet Transform and Image Registration	2-19
2.6 Summary	2-19

	Page
III. Methodology	3-1
3.1 Introduction	3-1
3.2 Wavelets and their Usefulness in Image Registration Algorithms	3-1
3.3 The New Multiscale Transform	3-2
3.3.1 The Lowpass Filter Design	3-3
3.3.2 The Continuous Wavelet Transform	3-5
3.3.3 The Two-Dimensional Multiscale Transform	3-7
3.4 Image Registration Algorithm Overview	3-9
3.5 Translation Algorithm	3-12
3.5.1 The Shift-Invariant Transform	3-13
3.6 Rotation Algorithm	3-14
3.6.1 Cartesian Coordinates to Polar Coordinates	3-15
3.6.2 The Rotation-Invariant Transform	3-16
3.7 Registration Algorithm Details	3-17
3.7.1 Subband Choice	3-17
3.7.2 Choosing Significant Coefficients	3-17
3.7.3 Masking	3-24
3.7.4 Initial Estimation	3-24
3.7.5 Final Estimation	3-26
3.7.6 The Multiscale Estimation	3-27
3.8 The Scale Algorithm	3-28
3.9 Summary	3-32
IV. Results	4-1
4.1 Introduction	4-1
4.2 Design of the Algorithm Validation Studies	4-2
4.2.1 Test Images	4-2

	Page
4.2.2 Peak Signal-to-Noise Ratio	4-3
4.2.3 The Validation Study	4-5
4.3 Translation Performance	4-10
4.3.1 Performance Results for Same Scale Registration	4-10
4.3.2 Performance Results for Different Scale Registration	4-20
4.4 Rotation Performance	4-23
4.4.1 Same Scale Rotation Registration Performance	4-24
4.4.2 Rotation Performance for Test Images at Different Scales	4-32
4.5 Summary	4-34
V. Discussion and Future Work	5-1
5.1 Contributions of this Thesis	5-1
5.2 Potential for Future Research	5-2
5.2.1 Implementation of a Fast Continuous Wavelet Transform Algorithm	5-2
5.2.2 Determining Appropriate Values for the Minimum Pearson Correlation	5-3
5.2.3 Creating an Algorithm to Register Scales . . .	5-3
5.2.4 Calculate Translation, Rotation, and Scale Changes Simultaneously	5-4
5.2.5 Creating a Super Resolution Algorithm	5-4
Bibliography	BIB-1
Vita	VITA-1

List of Figures

Figure		Page
2.1.	<i>The Two-Dimensional Discrete Wavelet Transform.</i>	2-12
2.2.	<i>The Frequency Response of the Two-Dimensional Redundant Discrete Wavelet Transform</i>	2-14
2.3.	<i>The Two-Dimensional Discrete Wavelet Transform</i>	2-15
2.4.	<i>The Two-Dimensional Continuous Wavelet Transform</i>	2-17
3.1.	<i>Lowpass Filter designs.</i>	3-5
3.2.	<i>Block diagram of the two-dimensional multiscale transform. .</i>	3-8
3.3.	<i>Scale two of the Redundant Multiscale Transform.</i>	3-9
3.4.	<i>One iteration of the redundant discrete wavelet transform compared to scale two of the multiscale transform.</i>	3-10
3.5.	<i>Converting Cartesian Coordinates to Polar Coordinates. . .</i>	3-16
3.6.	<i>Scale two of the Polar Multiscale Transform of the Cameraman Image.</i>	3-18
3.7.	<i>Example of the “zero frame” and “black-out” regions.</i>	3-20
3.8.	<i>Example of the “zero frame” region with no periodization for polar images.</i>	3-21
3.9.	<i>Example of the “zero frame” and “black-out” regions for polar images.</i>	3-23
3.10.	<i>Example of the masking algorithm.</i>	3-25
3.11.	<i>Finding the appropriate value for the minimum Pearson correlation.</i>	3-28
3.12.	<i>Comparison of significant coefficients of a signal and a lowpass filtered version of the signal.</i>	3-30
3.13.	<i>The alignment of the lowpass filters when the scales align. . .</i>	3-33
4.1.	<i>Cameraman and Lenna images.</i>	4-3

Figure		Page
4.2.	<i>Images used to test our multiscale registration algorithm when the reference and input images are at different scales.</i>	4-4
4.3.	<i>Examples of PSNR values for the test images.</i>	4-6
4.4.	<i>Translated versions of the test images.</i>	4-8
4.5.	<i>Rotated versions of the test images.</i>	4-9
4.6.	<i>Results for scale two of the multiscale translation algorithm compared to the results of a single iteration of the algorithm developed by Brown [5].</i>	4-12
4.7.	<i>Results for translation registration using scales two through six of the multiscale transform.</i>	4-15
4.8.	<i>Results for scale four translation registration using the multiscale transform compared to the results of the second iteration of the algorithm developed by Brown [5].</i>	4-16
4.9.	<i>Results for the multiscale registration algorithm for translation where the input and reference images are at the same scale. .</i>	4-18
4.10.	<i>Average scale used for each test image to register translation differences between two images when the images are at the same scale.</i>	4-20
4.11.	<i>Results for the multiscale registration algorithm for input and reference images at different scales.</i>	4-22
4.12.	<i>Average scale used for each test image to register the two images when the images are at different scales.</i>	4-23
4.13.	<i>Results for scale two rotation registration using the multiscale transform compared to the results of a single iteration of the algorithm developed by Brown [5].</i>	4-26
4.14.	<i>Results for rotation registration using scales two through six of the multiscale transform.</i>	4-27
4.15.	<i>Results for scale four rotation registration using the multiscale transform compared to the results of the second iteration of the algorithm developed by [5].</i>	4-29

Figure		Page
4.16.	<i>Results for the multiscale registration algorithm for rotation registration for images at the same scale.</i>	4-31
4.17.	<i>Average scale used for each test image to register the input image with rotation differences to the reference image when the images are at the same scale.</i>	4-33
4.18.	<i>Results for the multiscale registration algorithm for rotation registration where the input image and the reference image are a different scales.</i>	4-35
4.19.	<i>Average scale used for each test image to register the input image with a rotation difference to a reference image when the images are at different scales.</i>	4-36

Abstract

With recent advances in bandwidth, sensor resolution, and unmanned aerial vehicles technology, image data is being collected in larger quantities than ever before. All of this data must be analyzed in an accurate and efficient manner. Human analysis of large amounts of data is time consuming and inaccurate, so a fast, automated, accurate, and efficient method to register images is needed. Image registration is a key first step in the analysis of imagery data for the purpose of super-resolution and an integral part of algorithms used for target recognition, medical imagery, and computer vision. Once the images are registered, they can be compared and contrasted much more easily within these applications.

The most recent research involved registering images in the presence of translations and rotations using one iteration of the redundant discrete wavelet transform. We extend this work by creating a new multiscale transform to register two images with translation or rotation differences, independent of scale differences between the images. Our two-dimensional multiscale transform uses an innovative combination of lowpass filtering and the continuous wavelet transform to mimic the two-dimensional redundant discrete wavelet transform. This allows us to obtain multiple subbands at various scales while maintaining the desirable properties of the redundant discrete wavelet transform. Whereas the discrete wavelet transform produces results only at dyadic scales, our new multiscale transform produces data at all integer scales. This added flexibility improves registration accuracy without greatly increasing computational complexity and permits accurate registration even in the presence of scale differences. We demonstrate the performance of our algorithm by registering test images at various rotations and translations, in the presence of scale differences and additive white noise.

TRANSLATION AND ROTATION INVARIANT MULTISCALE IMAGE REGISTRATION

I. Introduction

1.1 Problem Statement

Imagery data of a single object can be collected from a variety of sensors as well as from several different positions. One example of this is the collection of intelligence data by an unmanned aerial vehicle (UAV). The UAV flies around the object, taking a large number of pictures from several directions, varying altitudes, and multiple sensors. Therefore, it is necessary to arrange these images in a way that allows easy interpretation. Image registration is one method that can aid in arranging the imagery data. Image registration is a key first step in the analysis of imagery data for the purpose of super-resolution and an integral part of algorithms used for target recognition, medical purposes, and computer vision.

Image registration involves taking a set of images and determining the necessary rotations, translations, scale changes, etc., needed to make a set of images align. Once registration is complete, the images can be compared and contrasted much more easily. Human analysis of the imagery data is one way to register images. However, this method is time consuming and inaccurate, especially when there are large amounts of data to be analyzed. Therefore, a fast, automated, efficient, and accurate method to register images is needed. The purpose of this research is to build a multiscale transform that can be used to approximate the translation and rotation differences between several images at varying scales in an accurate and efficient manner. This transform will be used to design a new multiscale image registration algorithm.

1.2 *Scope*

Previous work registered images in the presence of translations and rotations using one iteration of the redundant discrete wavelet transform. We extend this work by using the continuous wavelet transform at multiple scales to register images in the presence of scale changes in addition to translations or rotations. Since noise is a factor in data collection, the continuous wavelet transform alone cannot be used for registration due to its noise sensitivity. Our multiscale transform utilizes the continuous wavelet transform as well as a family of lowpass filters that approximate the Daubechies 9,7 discrete wavelet. This allows us to obtain multiple subbands at various scales while maintaining the desirable properties of the discrete wavelet transform. We demonstrate the performance of our algorithm by registering images at various scales, translations, and rotations in the presence of noise. An input image will be registered with a reference image to show the accuracy of this algorithm, and translations and rotations will be considered separately. However, this algorithm can easily be extended to register sets of multiple images.

Translations and rotations in an image should be detectable at any scale. Therefore, if an image has both translations or rotations and scale changes, our algorithm should remove the translation or rotation, despite the difference in scale. We will demonstrate the performance of our algorithm by registering pairs of images under benign and noisy conditions.

1.3 *Thesis Organization*

Chapter two contains background information on image registration and the wavelet transform. An overview of image registration and various registration techniques will be considered first. Following this discussion, there will be an overview of the wavelet transform with a focus on the continuous wavelet transform, its limitations, and its extension to two dimensions. The discrete wavelet transform will be discussed next as an introduction to the two-dimensional multiscale transform that

we have developed. The relevance of using lowpass filters in our new algorithm will be discussed. Finally, the above topics will be tied together with a discussion on using wavelet and multiscale transforms in image registration.

Chapter three will begin with a discussion on the wavelet registration concept followed by an in depth look at our image registration algorithm. First, the translation-invariant multiscale transform that can register translations at different scales will be discussed. Next the rotation-invariant transform will be considered. Finally, an algorithm to determine the scale differences between two images will be discussed.

In Chapter four, the performance of the registration algorithm will be demonstrated. This includes the validation of the new translation and rotation registration algorithms, design decisions, and the techniques that were developed to overcome issues that occurred during development.

In Chapter five, we will discuss our conclusions and propose avenues for future research.

II. Background

2.1 Introduction

The wavelet transform has become a useful tool in image registration algorithms [1, 4, 5, 9, 16, 18, 19, 20, 21, 23, 32]. This thesis extends the work of Brown [5] to incorporate a new multiscale transform into the problem of image registration. Thus, we begin our discussion with image registration. The properties that make wavelets useful to image registration will be presented next. A discussion of the continuous wavelet transform (CWT) and why it alone cannot be used for image registration will follow. Our alternate method to the CWT mimics the redundant discrete wavelet transform (redundant DWT). Thus, we will present some background information on the redundant DWT. The useful properties and drawbacks of the redundant DWT will be considered in this section to set the stage for the new multiscale transform that we have developed. Following this section, the lowpass filter will be discussed, which is the final element of our new multiscale transform. A section on the use of the wavelet transform in image registration will tie together the previous sections of this chapter.

2.2 Image Registration Overview

Image registration involves taking a set of images and determining the necessary rotations, translations, scale changes, etc., needed to make these images match. Once registration is completed, the images can be compared and contrasted much more easily. In general, image registration is used to align an input image to a reference image. Image registration is an important preprocessing step in the analysis of imagery data for the purpose of super-resolution, pattern recognition, medical imaging, and computer vision.

2.3 Elements of Image Registration

There are four key elements of image registration according to Brown [4]: the feature space, the search space, the search strategy, and the similarity metric.

1. Feature Space

The feature space is the collection of features that will be used to represent the image for matching. Examples of features include intensity of pixels, edges, contours, and line intersections. Choosing a good feature space is essential to create an accurate image registration algorithm [4].

Manjunath describes generality and robustness as key factors in identifying good features [26]. Different applications give rise to different significant features, so the selection of features must be general enough to account for the differences. A more general feature selection allows for a greater variety of uses for the registration algorithm. Robustness refers to the ability of the feature selection algorithm to choose the same features regardless of the transformations (scale changes, rotations, translations) that occur from one image to the next.

Li and Zhou have created guidelines to follow when selecting features from an image [23]. The first of these is that the selected features exist in the same location in both images, which is called consistency. The second guideline is that the feature points are spread evenly across the images. The features should be positioned in the areas of high contrast, and the points should be distinct in their surrounding areas.

2. Search Space

The search space is the set of transformations that will be used to make the two images match. For example, if the two images were misaligned by a translation alone, then the search space would include the set of all translations. Two common transformations are rigid body and affine. The rigid body transformation

includes at most a scale change, a translation, and a rotation, and an affine transformation is a rigid body transformation plus a shear [4]. A shear is a shift in an image where the points on opposite sides of a fixed line are shifted in opposite directions. This will slant the image and change its aspect ratio [4].

3. Search Strategy

The search strategy determines which transformation from the search space will be chosen to register the next set of images. Using the translation example from above, if the input image is translated compared to the reference image, the search strategy will be to search the set of all translations to find the appropriate translation to match the images [4].

4. Similarity Metric

Finally, the similarity metric is used to compute the degree of likeness of the two images. A common similarity metric is cross-correlation [4].

2.4 *Image Registration Techniques*

Several registration techniques have been developed over the years. In the past, these registration algorithms required some human interaction. Today, registration is moving toward automatic algorithms to register images due to the large amounts of data that can be collected in short periods of time. There are two categories of registration techniques that are considered here. These two methods are Fourier-based techniques [2, 7, 15, 30] and wavelet-based techniques [5, 9, 12, 16, 17, 18, 19, 21, 31, 32, 33]. Fourier-based algorithms are an important class of registration algorithms, and the precursor to wavelet techniques.

2.4.1 Fourier-Based Techniques. Fourier-based algorithms use the properties of the Fourier transform to register images. Shifts, rotations, scales, and other transformations can be represented in a compact, easily interpreted way in the

Fourier domain. The general approach of Fourier techniques is to use phase correlation, which exploits the translation property of the Fourier transform. If an input image is shifted when compared to a reference image, the only difference will be a phase difference, which can be used to determine the shift difference between the two images. Rotation is approximated in the same manner as translation, with the preprocessing step of changing the Cartesian coordinates to polar coordinates [4].

The Fourier method is relatively fast when using the fast Fourier transform (FFT). Another nice property of Fourier-based techniques is that they perform well, even with the addition of correlated and frequency-dependent noise. However, these techniques fail with the addition of white noise. Fourier techniques can handle, at most, rigid body transformations. But, they are independent of viewpoint, which makes them useful when registering images from different sensors [4].

The major shortcoming of the Fourier techniques is that the image is localized in frequency, giving no spatial information. This means that there is no way to determine at what position each frequency is located. Edges are high-frequency phenomena. Since edges are good candidates for the feature space, it would be nice to have some information about the location of the high frequencies in the Fourier transform. One method of creating frequency/space localization is windowing. The Short-Space Fourier Transform accomplishes this using a fixed window. However, there is a better way to accomplish the windowing method using wavelets [14], which will be discussed in the next section.

2.4.2 Wavelet-Based Techniques. Images are nonstationary signals, which makes them difficult to analyze using the Fourier transform because this transform describes a nonperiodic signal with a set of periodic functions [14]. The Fourier transform of the signal will give the frequency content of the entire image, without any indication of where those frequencies occurred. The Short-Space Fourier transform improves on the Fourier transform to some degree. This transform uses a fixed

window to compute the Fourier transform of the signal at points along a spatial scale. However, if the signal within that window has a longer period than the window itself the frequency information at that location cannot be resolved. If the signal has a short period when compared to the window, the time/space information cannot be resolved using this transform. The wavelet transform goes one step further by using a variable window, which allows the transform to better describe the two instances that the Short-Space Fourier transform cannot adequately describe [14].

Wavelet techniques for image registration are point-to-point matching techniques. This means that a set of points are taken from each image using some criteria, such as the maxima of the wavelet coefficients. These coefficients are computed by taking the wavelet transform of the signal. The points are matched to determine the transformation that is needed to align the images. This point-to-point matching technique is the most common practice today [21].

Wavelets have several nice properties that make them well suited for image registration. The maxima of the wavelet coefficients can be computed automatically, which means that the features can be computed without human interaction. This allows for the automation of image registration [16]. According to Tashakkori et al [35], the correlation between wavelet coefficients of a set of images is greater than the correlation between the set of images itself, which allows for a better prediction of the transformations needed to align the images. Finally, multiple scales of the wavelet transform can be used, which allows better registration accuracy because several scales can be utilized to create a more accurate prediction.

Crouse, et al [10] describe three primary properties and two secondary properties of wavelet transforms that make them a natural choice for image processing algorithms. The primary properties are locality, multiresolution, and compression, and the secondary properties are clustering and persistence. Locality refers to the idea that the wavelet is a “small wave” which has frequency and time information associated with it. The premise that the wavelet transform produces a decomposition

at several levels or scales refers to the property of multiresolution. The last primary property, compression, suggests that the wavelet transform reduces the amount of significant information to a small number of points. Clustering signifies that the coefficients that surround a large coefficient will tend to be large themselves, and small coefficients will tend to have small coefficients surrounding them. Finally, persistence means that wavelet coefficients that tend to be large at one scale will also be large at the other scales as well, and wavelet coefficients that are small at one scale will tend to be small at the other scales. These properties make wavelet techniques a natural choice for image registration; therefore, we will take a more detailed look at the wavelet transform in the next section.

2.5 The Wavelet Transform

The wavelet transform forms the basis of the new multiscale transform that will be used in our image registration algorithm. In particular, we will utilize a two-dimensional continuous wavelet transform and a lowpass filter which approximates the scaling function of the discrete wavelet transform. Previous work by Brown [5] shows that the two-dimensional redundant discrete wavelet transform works well for registering images with large translations or rotations. To create our new multiscale transform, the two-dimensional redundant discrete wavelet transform will be used as a guide. In order to set the background for our new transform, we must discuss several topics. This section will begin with a discussion on the continuous wavelet transform, followed by a discussion on the discrete wavelet transform. Next, the extension to two-dimensional wavelet transforms will be considered. Finally, there will be a brief discussion on lowpass filters that will aid in the understanding of our new multiscale transform.

2.5.1 The Continuous Wavelet Transform. The Continuous Wavelet Transform (CWT) has the form:

$$W_{\Psi}[f](b, a) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} \Psi\left(\frac{x-b}{a}\right) f(x) dx \quad (2.1)$$

where $f(x)$ is the continuous input signal, $\Psi(x)$ is known as the mother wavelet, a is the scale parameter, and b is the translation parameter. By varying the translation parameter at a given scale, the continuous wavelet transform can be interpreted as the details (high frequency components) of the input signal at the given scale. The mother wavelet and all of the translated and scaled versions of it are the spanning set that is used to describe the signal in the wavelet domain. The scale parameter allows for localization in scale (inverse of frequency), and localization in time is obtained using the shift parameter [14].

Matlab version 6 is used to compute the continuous wavelet transform. The discrete input signal is interpolated using piecewise constant interpolation. The continuous wavelet transform must be sampled in order to use it for digital signal processing. First, the wavelet function is approximated to a specified precision. Once again, a piecewise constant interpolation method is used. Then, the input signal and the approximation to the wavelet integral are correlated. Finally, the point-to-point difference is taken of the result of the correlation to form the wavelet coefficients. This algorithm keeps the L central points of the transform, where L is the length of the original signal. The CWT algorithm in Matlab version 6 works for one-dimensional signals only. In our two-dimensional transform, we operate along the rows and then along the columns in a separable fashion. Because of this, we can quickly take the continuous wavelet transform of the two-dimensional image. First, the signal is converted to a one-dimensional signal by concatenating along the rows. We take the transform of this signal and then convert the result back to a two-dimensional signal. Then, we take the transform along the columns by concatenating the signal

along the columns and taking the transform of the one-dimensional signal. Finally, we convert the signal back to two-dimensions. With this step, we have created the two-dimensional continuous wavelet transform of the original image.

Our new multiscale transform utilizes the continuous wavelet transform, but produces a transform that is similar to the discrete wavelet transform. To understand the new method that we have developed, it is necessary to briefly discuss the redundant discrete wavelet transform, and its nice properties that we will exploit in our design.

2.5.2 Function Space Development of the Discrete Wavelet Transform.

Discrete wavelet transforms work by projecting the input function into two smaller orthogonal subspaces. The first subspace is created from the shifts and dilations of a lowpass scaling function $\phi(x)$ and the second subspace is formed by the shifts and dilations of a highpass wavelet function $\psi(x)$ [11].

The set $\{V_m\}_{m \in \mathbb{Z}}$ is a sequence of nested subspaces in $L^2(R)$, where V_m is a subset of V_{m-1} for all $m \in \mathbb{Z}$, and f is a function in the subspace V_{m-1} for some m . P_m is the operator that projects f into the nested subspace V_m , which is a subset of V_{m-1} . P_m keeps the part of f that is contained in V_m and removes the rest. The set $\{W_m\}_{m \in \mathbb{Z}}$ is another sequence of nested subspaces in $L^2(R)$, where W_m is a subset of V_{m-1} for all $m \in \mathbb{Z}$. Let $Q_m = I - P_m$, where I is the identity operator, be the operator that projects f into the nested subspace W_m . Q_m preserves the part of f in W_m and removes the other parts of f . The span of $V_m \cup W_m$ equals V_{m-1} and V_m ; W_m are orthogonal subspaces. This means that $V_{m-1} = V_m \oplus W_m$, $V_m \cap W_m = \emptyset$, and $P_m Q_m = Q_m P_m = 0$, where 0 is the zero operator.

The following equations govern the performance of the discrete wavelet transform. For a fixed m , the set of scaling functions $\{\phi_{m,n}\}_{n \in \mathbb{Z}}$ are an orthonormal basis for V_m and the set of wavelet functions $\{\psi_{m,n}\}_{n \in \mathbb{Z}}$ are an orthonormal basis for W_m .

By expanding $\phi_{m,n}$ in terms of $\{\phi_{m-1,k}\}_{k \in \mathbb{Z}}$, the basis functions for the next higher subspace, we obtain the equation:

$$\phi_{m,n}(x) = \sum_k c_{m-1,n}(k) \phi_{m-1,k}(x). \quad (2.2)$$

The coefficients $c_{m-1,n}(k)$ are the coarse coefficients, which provide a coarse approximation to the input signal. These coefficients are found by taking the inner product of $\phi_{m,n}$ with $\phi_{m-1,k}$. That is, $c_{m-1,n}(k) = \langle \phi_{m,n}, \phi_{m-1,k} \rangle$. Now, if $h_n(k)$ is the inner product of $\phi_{m,n}$ with $\phi_{m-1,k}$, then by substituting for $c_{m-1,n}(k)$ in Equation 2.2 above, the equation becomes

$$\phi_{m,n}(x) = \sum_k h_n^m(k) \phi_{m-1,k}(x). \quad (2.3)$$

To create an orthogonal wavelet transform, the $\{\phi_{m,n}\}$ must satisfy Equation 2.3 for all m , that is, a single $h_n(k)$ satisfies Equation 2.3 for all m . Using the same steps from above with the $d_{m-1,n}(k)$, which are the detail coefficients, we obtain the following equation:

$$\psi_{m,n}(x) = \sum_k g_n(k) \phi_{m-1,k}(x). \quad (2.4)$$

where $g_n(k)$ is the inner product of $\phi_{m,n}$ with $\psi_{m-1,k}$, independent of m [6].

Other restrictions must be imposed to form the discrete wavelet transform. These restrictions are known as the wavelet recursion equations [6]. The wavelet recursion equations take the following form:

$$\begin{aligned} \phi(x - n) &= \sum_k h(k - 2n) \phi(2x - k) \\ \psi(x - n) &= \sum_k g(k - 2n) \phi(2x - k). \end{aligned}$$

where $h_n(k)$ has been replaced with $h(k - 2n)$ to reflect the dyadic nature of the transform.

The above discussion refers to orthogonal spaces. This places certain restrictions on the wavelet filters that are not desirable. By relaxing this condition, we can create biorthogonal wavelets, which are better suited for image registration, as will be discussed below. With biorthogonal wavelets, we have dual spaces. Thus, we now have V_m , W_m , \tilde{V}_m spanned by $\{\tilde{\phi}_{m,k}\}$, and \tilde{W}_m spanned by $\{\tilde{\psi}_{m,k}\}$. Now, $V_m \cap \tilde{W}_m = \emptyset$, $\tilde{V}_m \cap W_m = \emptyset$, and the basis functions must satisfy the biorthogonality condition shown below,

$$\begin{aligned}\langle \phi_{m,k}, \tilde{\phi}_{m,n} \rangle &= \delta(k - n) \\ \langle \psi_{m,k}, \tilde{\psi}_{m,n} \rangle &= \delta(k - n),\end{aligned}$$

where $\langle x, y \rangle$ is the inner product of x and y .

Because phase information is more important than magnitude information when reconstructing images, having linear phase is desirable [24]. By using biorthogonal wavelets, it becomes easier to design filters with linear phase. The biorthogonal wavelets can be made up of odd length filters unlike orthogonal wavelets. This allows symmetry which produces linear phase. Another improvement that biorthogonal wavelets have over orthogonal ones is that larger linear phase filters can be created, and larger filters correspond to smoother wavelet functions [34]. The only orthogonal linear phase filter is the Haar wavelet, which has a length of only two.

The discrete wavelet transform is a multiscale decomposition of a signal. At each scale, the signal is decomposed into a coarse approximation, $c_m(n)$, and a set of details, $d_m(n)$. At the next scale, the coarse approximation is broken down further into a more coarse approximation and another set of details. The filters h and g in the wavelet recursion equations for the discrete wavelet transform can be represented

in the following way,

$$\begin{aligned} h(k - 2n) &= \langle \phi_{m-1,n}, \phi_{m,n} \rangle \\ g(k - 2n) &= \langle \phi_{m-1,n}, \psi_{m,n} \rangle. \end{aligned}$$

By knowing the coefficients $c_{m-1}(n)$ of $f \in V_{m-1}$, we can project f into V_m and W_m using the following equations,

$$\begin{aligned} c_m(n) &= \sum_k c_{m-1}(k) h(k - 2n) \\ d_m(n) &= \sum_k c_{m-1}(k) g(k - 2n). \end{aligned}$$

Using the above equations, the signal can be decomposed further into V_{m-1} and W_{m-1} . For further information on the discrete wavelet transform see references [6, 8, 11, 25, 34]. Now, we extend the wavelet transform to two dimensions.

2.5.3 The Two-Dimensional Wavelet Transform. An image is a two-dimensional signal, and therefore is best analyzed by a method other than a one-dimensional transform. The two-dimensional wavelet transform processes the signal by first performing a transform along the rows, and then transforming the columns of the new signal. This type of process is known as separable processing. In this section, we will discuss the two-dimensional discrete wavelet transform and the two-dimensional continuous wavelet transform.

2.5.3.1 The Two-Dimensional Discrete Wavelet Transform. The two-dimensional discrete wavelet transform decomposes the image into four sub-bands, each of which is reduced by a factor of two along both the rows and the columns. This type of decomposition is called a dyadic decomposition because the data is reduced in size by a factor of two from scale to scale in a one-dimensional sense. In two-dimensional processing, the signal is reduced by a factor of four because

the data size is reduced by two along the rows and along the columns. Figure 2.1(a) shows the original image (at scale zero), and Figure 2.1(b) is the two-dimensional discrete wavelet transform of the original image after the first decomposition of the data (scale one).

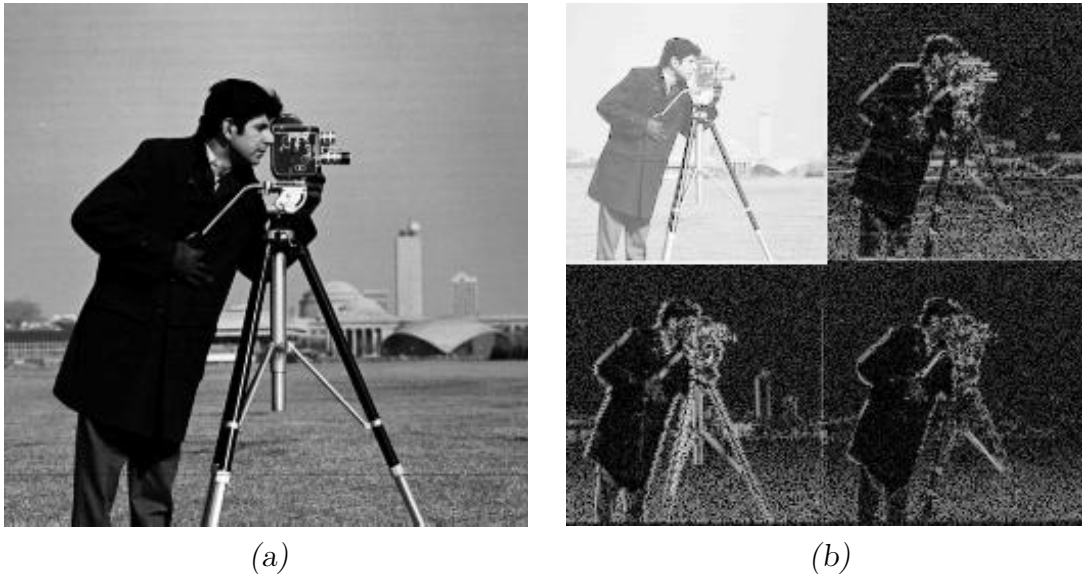


Figure 2.1. *The Two-Dimensional Discrete Wavelet Transform. (a) The Original Cameraman Image. (b) One iteration of the discrete wavelet transform. The LL subband is in the upper left. The LH subband is in the upper right. The HL subband is in the lower left, and the HH subband is in the lower right.*

The two-dimensional DWT is analogous to the one-dimensional DWT in that there is a coarse approximation of the original signal followed by a series of details. The top left quadrant of Figure 2.1(b), known as the low-low (LL) subband, is the coarse approximation to the signal. This subband is created by first convolving the rows of the image with the $h(n)$ (see Section 2.5.2) and all of its shifts, and then convolving the columns of the result with $h(n)$ and its shifts. This, in effect, involves lowpass filtering along the rows and then lowpass filtering along the columns. The other subbands will follow the same convention as this one, where the first letter of the subband, in this case L, will refer to the processing along the rows, and the second letter of the subband, which is L as well, will refer to the processing along

the columns. This LL subband is the coarse approximation of the image. This approximation is essentially a blurring of the image in both directions, since that is the effect of the lowpass filter on an image. The LL subband of the transformed image is processed to obtain the next level of decomposition (scale two).

The other three subbands at scale one in the transformed image are referred to as the detail subbands. The subband in the upper right of the transformed image is the low-high (LH) subband. This subband is created by first convolving the rows of the image with $h(n)$, and then convolving the columns of the result with $g(n)$ (see Section 2.5.2). Since $g(n)$ acts like a highpass filter, the LH subband involves lowpass filtering the rows, followed by highpass filtering the columns. This results in the preservation of horizontal edges. The highpass filter preserves high frequency phenomena, which correspond to the edges.

The high-low (HL) subband is located in the bottom left of the transformed image, as shown in Figure 2.1(b). This subband is composed of a convolution of $g(n)$ with the rows of the original image followed by a convolution of $h(n)$ with the columns of the resulting image. This portion is highpass filtered along the rows and lowpass filtered along the columns, which preserves the vertical edges and blurs the data along the columns.

The high-high (HH) subband is in the bottom right of Figure 2.1(b). This part of the transformed image was created by highpass filtering the rows of the image and then highpass filtering the columns of the resulting image. This subband preserves the details that are manifested as edges at a 45 degree angle in either direction from the vertical.

To get a better understanding of how these subbands are formed, we take a look at the frequency response of the filters used to create each subband. Figure 2.2 shows the two-dimensional frequency response of each subband. In Figure 2.2(a), only the low frequencies in both directions are passed to achieve the LL subband. This corresponds to the white square in the center of the image. The black area

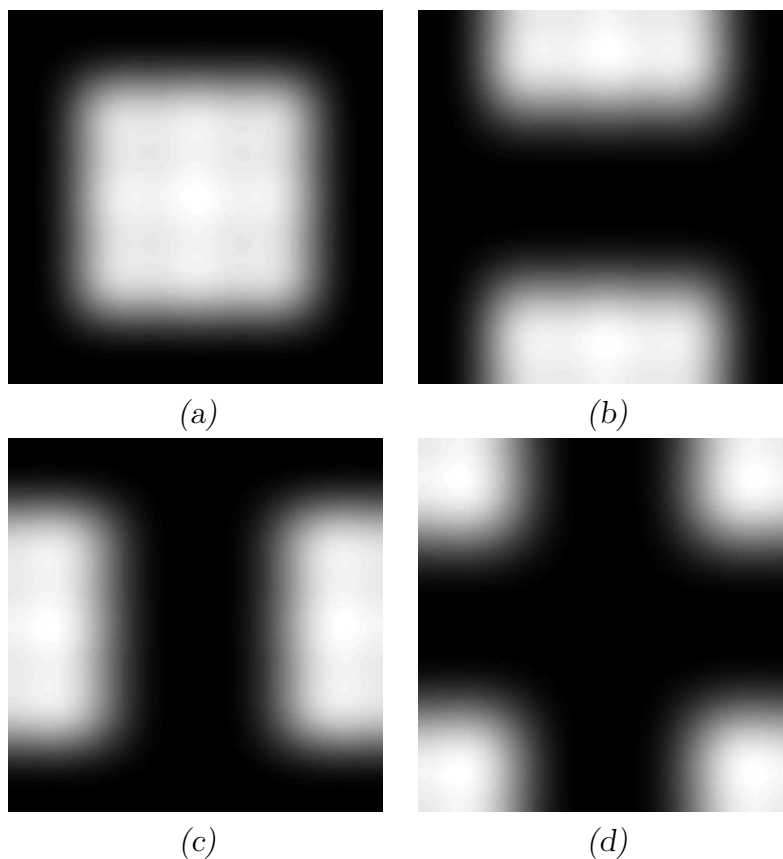


Figure 2.2. *The Frequency Response of the Two-Dimensional Redundant Discrete Wavelet Transform. (a) The frequency response of the filter that produced the LL subband. (b) The frequency response of the filter that produced the LH subband. (c) The frequency response of the filter that produced the HL subband. (d) The frequency response of the filter that produced the HH subband.*

corresponds to the frequencies that were not passed using this filter. Figure 2.2(b) shows the LH subband. The high frequencies of the rows are not passed, which corresponds to the areas of black at the left and right of the figure. Similarly, the low frequencies of the columns were not passed in this filter, which corresponds to the horizontal patch of black in the center of the image. Figure 2.2(c) and (d) can be analyzed in the same way.



Figure 2.3. *The Two-Dimensional Discrete Wavelet Transform. There are three iterations of the two-dimensional discrete wavelet transform shown. With each iteration, the number of pixels decreases by a factor of four, when using the discrete wavelet transform. We will be using redundant wavelet transforms, which remain the same size with each iteration. This image shows that the LL subband of the previous scale is processed in each iteration of the wavelet transform to produce a LL, LH, HL, and HH subband at the next scale.*

The discrete wavelet transform is a multiscale transform. Figure 2.3 shows a pictorial view of the decomposition that the discrete wavelet transform produces. The LL subband from the previous scale is processed to create the next scale of

the transform. This decomposition can be taken to any dyadic scale, but we must remember that each preceding scale is reduced by a factor of four at the next scale. Thus, at some point the data becomes too small to be reduced any further, and at this point the next scale cannot be found. To correct this problem, a redundant discrete wavelet transform can be used. This transform preserves the size of the data going from scale to scale. The redundant wavelet transform is also shift-invariant, a key property lacking in the DWT. This makes the two-dimensional redundant discrete wavelet transform ideal for image registration with some exceptions. The two-dimensional redundant discrete wavelet transform is not scale-invariant. Ideally, we wish to produce one transform that can be used to register images with translations, rotations, and scale changes. Without scale-invariance, we cannot determine the scale of the images compared to a reference. Also, the redundant discrete wavelet transform can only produce data on dyadic scales, which means that the scales between these dyadic scales cannot be utilized. We wish to use the redundant discrete wavelet transform concept to produce a new multiscale transform. In order to do this, we will use the continuous wavelet transform, which will be discussed next in the two-dimensional sense.

2.5.3.2 Two-Dimensional Continuous Wavelet Transform. The two-dimensional continuous wavelet transform is created by taking the continuous wavelet transform along the rows of an image, and then taking the continuous wavelet transform along the columns of the resulting image. The CWT does not have an equivalent to the scaling function of the DWT, and thus the lowpass and bandpass portions of the two-dimensional discrete wavelet transform cannot be produced. See Figure 2.4. Using the continuous wavelet transform in two dimensions produces only an equivalent to the high-high subband of the two-dimensional discrete wavelet transform. Thus, the continuous wavelet transform cannot be used alone for real-world image registration due to the noise sensitivity of this subband [32]. If we want to use a continuous wavelet transform approach to the registration problem, we must find a

way to produce the subbands that would be less sensitive to noise. In order to do this, we have chosen to create a transform similar to the redundant discrete wavelet transform. To produce the bandpass subbands, we need some sort of lowpass filtering step to reduce the noise in the signal. In the next section, we will discuss the lowpass filters that will be used as the $h(n)$ of our new multiscale transform.

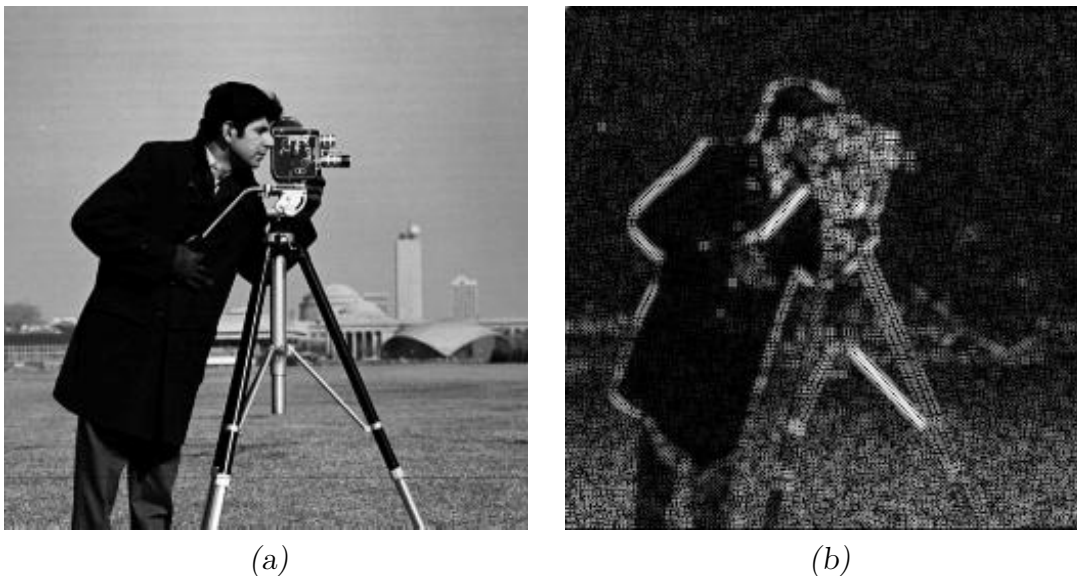


Figure 2.4. *The Two-Dimensional Continuous Wavelet Transform. (a) The original image. (b) Scale two of the Two-Dimensional Continuous Wavelet Transform. The two-dimensional continuous wavelet transform produces an image that has been highpass filtered along both the rows and the columns. This corresponds to the HH subband of the redundant discrete wavelet transform.*

2.5.4 Lowpass Filters. It is important to entertain a brief discussion on lowpass filters. This sets the stage for the development of a wavelet transform in which the highpass portion of the transform is based on the continuous wavelet transform, and the lowpass portion of the transform is created by a lowpass filter that mimics the $h(n)$ of the discrete wavelet transform.

In our design, we will use linear-phase FIR filters. A linear phase filter has filter coefficients that have either an even or odd symmetry. These filters have restrictions on their frequency responses depending on whether this symmetry is even or odd, and

on whether the length of the filter is even or odd. The symmetry of our coefficients will be even, and the filter order will be odd (Type two filter). For Type one filters, there are no restrictions on the frequency response, but for Type two filters, the frequency response at the Nyquist must be zero [36].

There are several methods commonly used to design lowpass filters. Three of these methods will be discussed here. The first is the Parks-McClellan optimal equiripple FIR filter design. These filters are designed so that the filter best approximates the given response in a minimax sense. This means that it minimizes the maximum error between the designed filter and the ideal response. This method uses the Parks-McClellan algorithm, which utilizes the Remez exchange algorithm and the Chebyshev approximation theory [36].

The second method of Linear-phase FIR filter design uses least-squares error minimization. The integral squared error between the desired frequency response of the filter and the actual frequency response of the design is minimized in this filter design. Both the equiripple filter and the least-squares error minimization filter include transition regions in their design which are considered “don’t care” regions, and thus allow a better response in the passband and stopband of the filter. The tradeoff of this is that the filter transitions from the passband to the stopband at a slower rate. The least-squares error minimization linear-phase FIR filter design concentrates the error at the edges of the passband and stopband of the filter, while the equiripple filter design spreads the error evenly over the entire passband and stopband [36].

The third method is the FIR filter design using the window method. Windowing involves applying a window to the inverse Fourier transform of the desired ideal filter, which truncates the filter. This design method finds the best filter in the integrated least squares sense by using a square window [28]. However, square windows lead to excessive ripple. To remove this ripple, many other windows are available, but they are not optimal and lead to increased transition bands.

2.5.5 The Wavelet Transform and Image Registration. Previous work by Brown [5] has developed a translation-invariant wavelet transform and a rotation-invariant wavelet transform to be used in an image registration algorithm. Brown found that the LL subband is not conducive to image registration because the transform coefficients are not sparse. Also, the HH subband is not a good candidate for image registration because it is sensitive to noise. Therefore, the best results in the Brown image registration algorithm are obtained using the LH and the HL subbands of the transform [5].

From the above discussion, we note that the continuous wavelet transform contains information that is similar to the HH subband of the discrete wavelet transform, and therefore due to its noise sensitivity is not a good candidate for image registration. Therefore, our goal is to create a lowpass filter portion to use with the continuous wavelet transform. With these two components, we hope to mimic the LH and HL subbands of the DWT to resolve images at different scales, along with rotations and translations, and obtain a transform that will be able to register all rigid body transformations of images.

2.6 Summary

This chapter began with a discussion on image registration and the properties that were necessary to obtain a good image registration method. Some background was given on several techniques used for image registration including Fourier transform registration techniques and Short-Space Fourier Transform techniques, which are the precursors to the wavelet techniques that will be used in the current registration algorithm. The continuous wavelet transform and wavelet transform properties that make this transform ideal for image registration were considered next. The discrete wavelet transform and the two-dimensional wavelet transform were a necessary part of the background to understanding the new transform that will be developed in Chapter III. Finally, the lowpass filter descriptions will allow for an easier under-

standing of the design decisions made about the lowpass portion of the new transform developed in the next chapter.

III. Methodology

3.1 Introduction

Wavelet transforms can be used to accurately predict the rigid body transformations needed to make two images align. In Chapter II, we discussed why wavelet transforms are useful in image registration algorithms. Our new algorithm can predict the translations or rotations needed to register two images independent of the scale changes that occur between the two images. To do this, the continuous wavelet transform will be used with a lowpass filter to create the LH subband and the HL subband, which have previously been shown to produce the most accurate registration results [5]. In this chapter, this new multiscale transform will be explained along with how it is used to register images that have translation or rotation differences in the presence of scale changes.

3.2 Wavelets and their Usefulness in Image Registration Algorithms

In Chapter II, we discussed several characteristics that are important for creating a robust image registration algorithm. The first requirement is to select an adequate feature space. In order to do this, several guidelines were put forth. The coefficients produced by the wavelet transform (wavelet coefficients) satisfy many of these guidelines. As discussed previously, the wavelet coefficients essentially define the edges in an image. These edges will correspond to the same features in the two images to be compared, thus satisfying the property of consistency. The wavelet coefficients exist where the edges are located and therefore should be spread throughout the image, which is the second property given by Li and Zhou [23]. The edges in the image by definition are the areas of high contrast, which satisfies the third property described by Li and Zhou. The wavelet coefficients are general, in that these coefficients will detect the edges in an image regardless of what type of image is used as input into the algorithm. Various images such as Infrared images, SAR images,

and video images can be used as inputs, and the edges would be detected in the same manner. Finally, the same features (the edges) will be chosen regardless of the transformations (scale changes, rotations, translations) that occur between the two images. This is the case for two reasons. First, we have developed a shift-invariant transform. Secondly, the significant coefficients (edges) persist across scale [27]. Now that we have shown that wavelet transforms satisfy many of the properties needed to obtain good image registration, we will discuss our new multiscale transform and how it utilizes the wavelet transform.

3.3 The New Multiscale Transform

A design requirement of our new multiscale transform is to mimic the performance of the redundant discrete wavelet transform (DWT). As mentioned in Section 2.5.5, the LH and HL subbands of the DWT are key for image registration due to their robustness in the presence of noise. The lowpass portions of these subbands correspond to the scaling function of the DWT, as described in Section 2.5.2. Unfortunately, the continuous wavelet transform (CWT) does not utilize a scaling function. Thus, we cannot create the needed subbands using only the CWT. To remedy this issue, we construct a new multiscale transform which produces these subbands using the CWT in combination with a lowpass filter.

In order to continue this discussion, we note the numbering scheme of the multiscale transform scales and contrast them with the redundant DWT scales. The first scale of the DWT (one iteration) results in a decomposition into two orthogonal subspaces, as described in Section 2.5.2. The basis functions for these subspaces are formed by “stretching” the basis functions of the original space by a scaling factor of two. Therefore, we refer to this scale of the CWT as scale two. Likewise, scale two of the DWT (two iterations) results in a further scaling by a factor of two. At scale two, we have a total scaling of four from the original basis functions. Thus,

we refer to this scale of the CWT as scale four. Further iterations are named in a similar fashion.

Our goal is to develop a multiscale transform based on the CWT that mimics the performance of the DWT. At each of the dyadic scales of the CWT (two, four, eight, etc.), the DWT utilizes half-band, quarter-band, and eighth-band filters (at DWT scales one, two, and three, respectively). These filters increase in total size (order) by a factor of two at each scale. Thus, to simulate the subbands of the DWT at these scales, we combine the CWT with an appropriately designed filter with a suitable cutoff. For instance, at scale two of the CWT, we must design a half-band lowpass filter to use as the lowpass portion of the multiscale transform.

With this multiscale transform, we are attempting to generate the integer scales between the dyadic scales of the DWT. We intend to utilize these CWT scales (three, five, six, seven, nine, etc.), which have no equivalent within the DWT framework. Thus, there is not a “target” lowpass filter to drive our design specification at non-dyadic scales. However, by logical extension, we determine that the filter choice at CWT scale N is an N^{th} -band filter, that is, a lowpass filter with cutoff frequency $\frac{2\pi}{N}$. Thus, our lowpass filter design problem becomes one of designing N^{th} -band filters for each scale N of the CWT, with the added constraint that they mimic the performance of the DWT lowpass filters at dyadic scales.

3.3.1 The Lowpass Filter Design. We wish to mimic the Daubechies 9,7 wavelet recursion equation lowpass filter ($h(n)$) with our designed lowpass filter. We chose this wavelet due to observations made by Brown [5]. Brown determined that the Daubechies 9,7 wavelet is an excellent feature extractor that has an outstanding ability to reduce noise while creating a parsimonious signal representation.

We wish to mimic the wavelet recursion equation lowpass filter at the dyadic scales. However, we would also like the lowpass filter creation to be automatic within our multiscale transform algorithm. It would be possible to design the quarter-band

filter by matching it to the equivalent scale two $h(n)$ filter, and so on for all dyadic scales. Still, for the “non-power-of-2” scales, there is no exact equivalent $h(n)$ filter in the DWT, since the redundant DWT only provides data at dyadic scales. Thus, we choose a design method that is consistent for all CWT scales. This process still closely represents the scale two $h(n)$ filter of the Daubechies 9,7 wavelet for the redundant DWT. Therefore, we will design a half-band lowpass filter to mimic the scale one wavelet recursion equation lowpass filter of the Daubechies 9,7 wavelet and make adjustments to this filter for the remaining scales.

We designed several lowpass filters to mimic the Daubechies 9,7 $h(n)$ half-band lowpass filter by varying the width of the transition band as well as varying the location where the transition band begins using the three design methods of Section 2.5.4. The best design for each of the three algorithms was determined by two methods, both of which compare the frequency response of the designed lowpass filter to the frequency response of the Daubechies 9,7 wavelet recursion equation lowpass filter. The first method used to determine the best filter design involved minimizing the maximum error between the frequency response of the designed filter and the frequency response of the $h(n)$ of the Daubechies 9,7 wavelet. The second method involved minimizing the sum of the absolute error between the frequency response of the designed filter and the frequency response of the $h(n)$ of the Daubechies 9,7 wavelet. Using these two error minimization techniques with the three lowpass filter designs gives us six lowpass filters. These lowpass filters were compared to each other, and the lowpass filter that most closely resembled the frequency response of the $h(n)$ was chosen. With both error minimization methods, the linear-phase FIR filter using least-squares minimization most closely approximated the $h(n)$. Figure 3.1 shows the two best lowpass filters as determined by the two error minimization methods. Each filter is shown with the Daubechies 9,7 $h(n)$ filter. Figure 3.1(b) shows the filter design that was chosen as the lowpass filter for our algorithm.

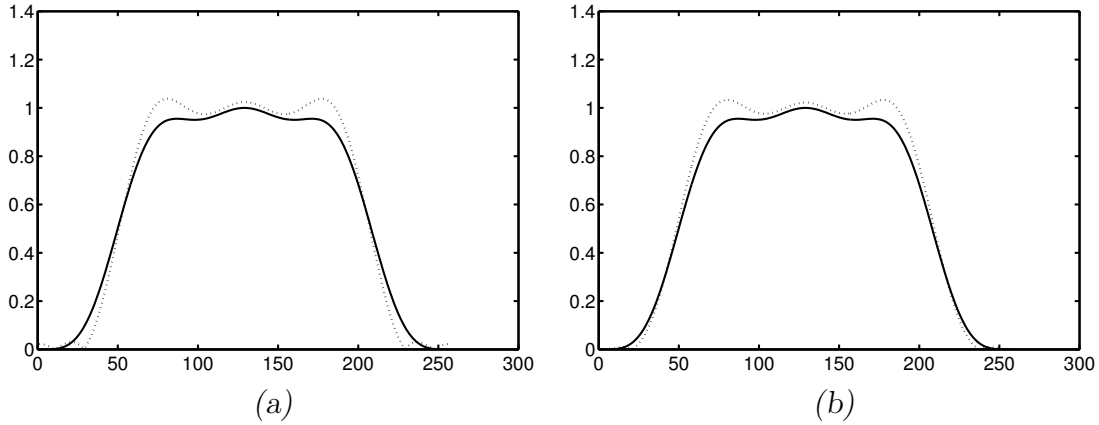


Figure 3.1. *Lowpass Filter designs. (a) The dashed line shows the Linear-phase FIR filter design using least-squares error minimization, determined by minimization of the maximum error. (b) The dashed line shows the Linear-phase FIR filter design using least-squares error minimization, determined by minimization of the sum of the errors. The solid line in (a) and (b) is the Daubechies 9,7 $h(n)$ filter which we are trying to approximate.*

To design the remaining N^{th} -band filters, the cutoff frequency is $\frac{2\pi}{N}$, as mentioned previously. The number of coefficients is increased appropriately at each scale, so that the number of coefficients at dyadic scales of the CWT correspond to the number of coefficients in the $h(n)$ of the DWT. The width of the transition band will be obtained by dividing the initial transition band by half of the current scale. For instance at scale three, if the transition band length for the half band filter were 0.3, the transition band for the third-band filter would be $0.3 \times \frac{2}{3}$. This method was chosen to automate the process of finding the transform for the scales beyond scale two. At this point, we have completely described the lowpass filter (the lowpass portion of this new transform), so we now turn our focus on the continuous wavelet transform (the highpass portion of the transform).

3.3.2 The Continuous Wavelet Transform. The continuous wavelet transform (CWT) replaces the $g(n)$ (wavelet recursion equation highpass filter of the redundant DWT) in our new multiscale algorithm. The continuous wavelet trans-

form was chosen so that integer scales can be obtained between the dyadic scales of the redundant DWT. By using the CWT, our algorithm can be easily extended to utilize non-integer scales if necessary. In the past, the continuous wavelet transform has not been utilized because it has a very slow computation time. However, several algorithms have been developed to implement the continuous wavelet transform in a fast manner on integer scales [3, 22, 29, 37, 38]. Within our algorithm, the CWT function in Matlab version 6 was used to compute the continuous wavelet transform. However, since we restrict ourselves to only use integer scales, a fast implementation of the continuous wavelet transform can easily be inserted into this algorithm in place of the current CWT implementation.

One of the goals of our multiscale registration algorithm is to provide fast registration. Normally, we would process the rows one by one. However, this causes the computation of the CWT to be very slow. Therefore, we chose to vectorize the two-dimensional image to form a one-dimensional signal and then transform this signal with the CWT algorithm. In order to do this, we must make some adjustments to the data. When the image is reshaped to form the two dimensional image again, border effects are apparent and must be removed. The border effects are caused when there are low pixel values on the end of one row, and high pixel values at the beginning of the next row, or vice-versa. Within our CWT algorithm, these pixels would be next to each other.

The first step of the CWT algorithm is to correlate the signal with the approximation to the wavelet integral. The regions with similar pixel values will tend to produce a low value in this correlation, and areas of transition from low to high pixel values will tend to produce a high value in this correlation. Large wavelet coefficients formed in this way typically correspond to edges in the image. However, when we are moving from one row to the next in the vectorized image, we may have an abrupt transition from high to low pixel values, due to the wavelet function extending beyond the border of the first row. Large wavelet coefficients formed in

this manner should not be considered as significant coefficients, and therefore must be removed.

In order to eliminate the border effects, an area ten pixels in width along the four borders of each subband are zeroed out. This frame of zeros ensures that border effect coefficients are not chosen as significant by the algorithm. The ten pixel width of the frame was experimentally determined by computing the multiscale transform at scales two through six.

3.3.3 The Two-Dimensional Multiscale Transform. Our two-dimensional multiscale transform uses a lowpass filter and the continuous wavelet transform to mimic the redundant discrete wavelet transform. Figure 3.2 shows how each subband of the multiscale transform is computed. Currently the multiscale transform is computed at integer scales, but the algorithm can easily be modified to compute the transform at non-integer scales. A representation of one iteration of the multiscale transform is shown in Figure 3.3. Figure 3.3(a) shows the Cameraman image and Figure 3.3(b) shows scale two of the multiscale transform of the Cameraman image. The image in the upper left of Figure 3.3(b) shows the LL subband of the transform. This subband is formed by lowpass filtering the rows. Then, the columns of the resulting image are lowpass filtered. The upper right image is the LH subband. This subband is created by first lowpass filtering the rows, and then taking the continuous wavelet transform of the columns of the resulting image. The image in the lower left results from taking the continuous wavelet transform of the rows, followed by lowpass filtering the columns of the resulting image. This image is called the HL subband. The image in the lower right corner is the HH subband. This image is formed by taking the continuous wavelet transform along the rows and then along the columns of the resulting image. To compute the scales beyond scale two of the multiscale transform, we iterate using the original image and not the LL subband of the previous scale, as we would if using the redundant DWT. We must iterate on the original image because we perform the multiscale transform on integer scales.

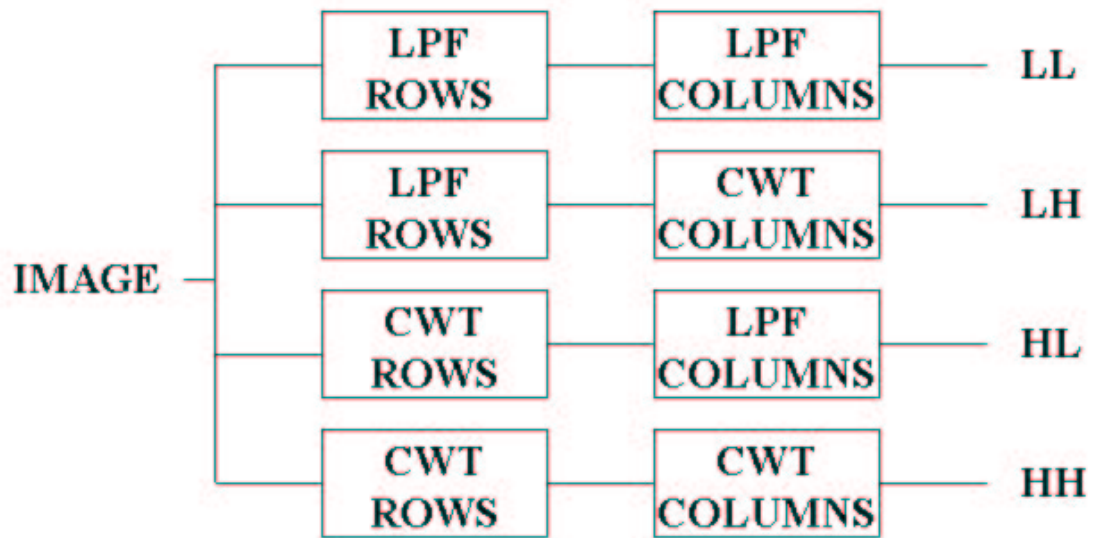


Figure 3.2. *Block diagram of the two-dimensional multiscale transform. This diagram shows how each subband of the multiscale transform is computed. The LL subband is computed by lowpass filtering along the rows and lowpass filtering along the columns. The LH subband is computed by lowpass filtering the rows and taking the CWT along the columns. The HL subband is computed by taking the CWT along the rows and lowpass filtering along the columns. The HH subband is computed by taking the CWT of the rows and then taking the CWT of the columns.*

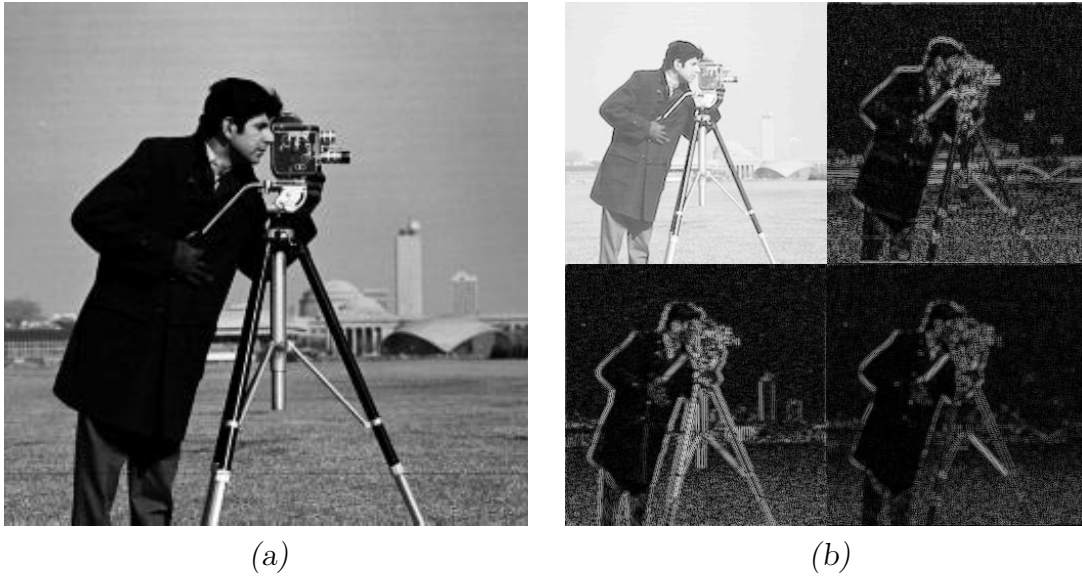


Figure 3.3. *Scale two of the Redundant Multiscale Transform. (a) The original image. (b) Scale two of the Redundant Multiscale Transform. The upper left image shows the LL subband, the upper right image shows the LH subband, the lower left image is the HL subband, and the HH subband is in the lower right.*

If we compare the redundant discrete wavelet transform in Figure 3.4(a) to the new multiscale transform of Figure 3.4(b), we see that the transforms look very similar. This is by design; we required our transform to mimic the DWT at dyadic scales. The multiscale transform improves on the redundant DWT because the multiscale transform can be utilized at any scale, whereas the redundant DWT is restricted to dyadic scales.

3.4 Image Registration Algorithm Overview

Our image registration algorithm can register images which are shifted or rotated when compared to a reference image independent of whether the two images are at different scales. This algorithm is a multiscale algorithm which utilizes as many scales as necessary to accurately predict a translation or rotation to a percentage Pearson correlation specified by the user.

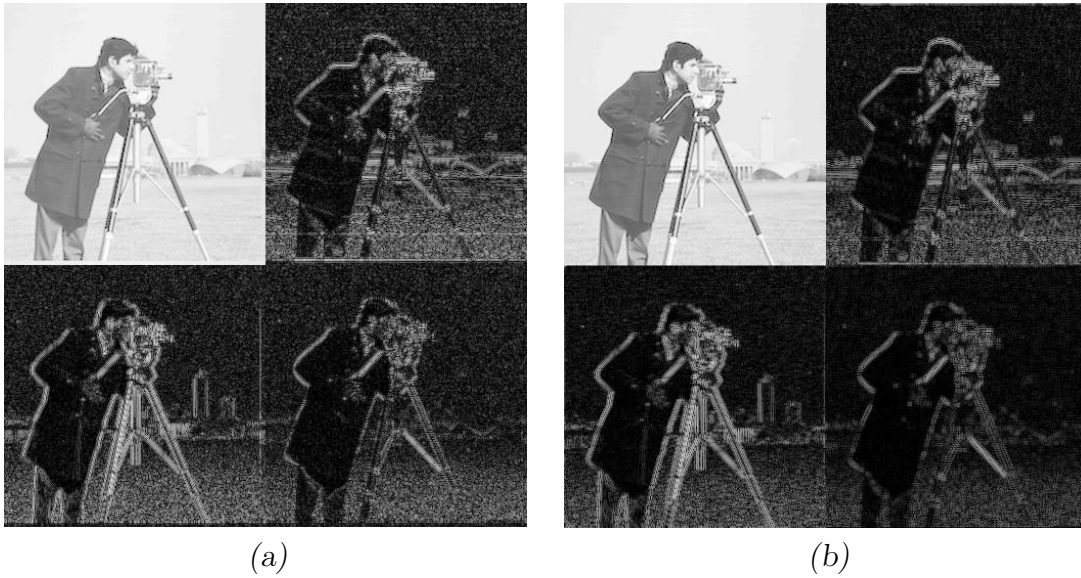


Figure 3.4. *One iteration of the redundant discrete wavelet transform compared to scale two of the multiscale transform. (a) One iteration of the redundant discrete wavelet transform. (b) Scale two of the multiscale transform. In (a) and (b), the upper left image shows the LL subband, the upper right image contains the LH subband, the lower left image is the HL subband, and the HH subband is in the lower right.*

When we register an input image to a reference image where the two images are at different scales, we consider the scales to represent different altitudes. Suppose we have two images. The first image is taken at an altitude greater than the second image. If these two images were taken using the same sensor, the object in the first image would be much smaller than the object in the second image. In order to register these images, the two images must represent similarly sized objects. Therefore, the first image will have to be cropped and enlarged to the size of the second image. When performing this enlargement, there will be points in between the known pixels (assuming digital data) from the original image that will be unknown. These unknown points must be interpolated. Clearly, the first image will be band-limited, and will be less “crisp” than the second image; the images are at different scales. This enlargement is considered a preprocessing step and is not part of this algorithm. Therefore, we assume that we start with images of objects of the same size, where one is more “blurry” (at a coarser scale) than the other.

In addition to registering images that are translated or rotated compared to a reference, we would also like to determine the scale difference between the reference and the input images. A goal of this registration algorithm is to create a scale-invariant wavelet transform. We analyze two images at different scales (altitudes) by taking the multiscale transform of the two images to determine the scale of the input image relative to the reference image. Although preliminary results are promising, this aspect of our algorithm still provides sporadic estimates. Therefore, the methods used to create this scale-invariant transform will only be discussed briefly.

In the next sections, the multiscale registration algorithm will be presented. There are three parts to this registration algorithm: the translation algorithm, the rotation algorithm, and the scale algorithm. The translation and rotation algorithms are similar in many ways. Therefore, the translation algorithm will be discussed along with elements that are unique to this algorithm. Next, the rotation algorithm will be discussed with the details unique to its design. Then, the common elements in

the design of these two algorithms will be discussed. Finally, we will briefly discuss the scale-invariant registration algorithm.

3.5 Translation Algorithm

Our translation algorithm can predict translation changes to a given degree of accuracy specified by the user. This algorithm is similar to that constructed by Brown [5]. The steps of this algorithm are shown below:

1. The user specifies the input image, the reference image, the minimum Pearson correlation desired, and the number of significant coefficients (N). The lowpass filter can be specified completely, or it can be chosen using one of the three lowpass filter design methods described in Section 2.5.4. The number of scales (S) that the user wishes to test must also be specified. The default is $S = 6$.
2. The multiscale transform of both the input image and the reference image are computed at scales two through S .
3. The LH subband and the HL subband are extracted from the wavelet transform data, and the significant coefficients at the first scale (scale two) are determined.
4. The coefficients are masked, where ones are placed in the locations of the significant coefficients, and zeros are placed elsewhere.
5. An initial estimate is obtained by comparing the 1st, 10th and middle significant coefficients of the LH subband of the reference image with all of the significant coefficient locations of the LH subband of the input image. This results in $3N$ initial estimates of the translation. The same operation is completed using the HL subband, producing another $3N$ estimates.
6. The masked LH subband of the reference image is then shifted by each of the $3N$ estimates for the LH subband. The masked LH subband of the reference image and the masked LH subband of the input image are correlated for each of

the translation estimates for that subband. The translation of the LH subband that has the best correlation is taken as the proper translation for that subband. This process is repeated for the HL subband, From this step, we have the LH subband translation estimate and the HL subband translation estimate.

7. In this step, the reference image is shifted by each of the two subband estimates. The Pearson correlation is computed using the input image and the shifted reference image for the two subband estimates. The estimate with the best correlation is taken as the correct translation.
8. If the Pearson correlation value does not meet the user specified minimum Pearson correlation required, the algorithm will begin again at step three using the values at the next highest scale.
9. Translate the input image to the reference image based on the solution to step seven once the minimum Pearson correlation requirement is met. If the minimum Pearson correlation requirement is not met, the highest correlation achieved will be chosen.

3.5.1 The Shift-Invariant Transform. Our shift-invariant wavelet transform is a two dimensional wavelet transform. A lowpass filter was chosen by finding the best fit to the Daubechies 9,7 discrete wavelet transform. The continuous wavelet transform is used to create the detail subbands. Figure 3.2 shows how each of the four subbands are formed. The rows in each image are analyzed first. To create the LL subband, first we take the image and lowpass filter along the rows. Then, the resulting image is lowpass filtered across the columns. To create the LH subband, the image is lowpass filtered along the rows, and then the continuous wavelet transform is taken of the resulting image along the columns. Similarly, to create the HL subband, the continuous wavelet transform is taken along the rows, followed by lowpass filtering the columns of the resulting image. For the HH subband, the continuous wavelet transform is taken along the rows of the original image. Then,

the continuous wavelet transform is taken along the columns of the resulting image. The order of the operations to form the subbands does not matter due to linearity of the lowpass filter and the CWT. For example, we could take the CWT of the columns followed by lowpass filtering the rows of the resulting image to form the LH subband if we prefer.

3.6 *Rotation Algorithm*

Our rotation algorithm can predict rotation changes to a given degree of accuracy specified by the user. The steps to this algorithm are shown below:

1. The user specifies the input image, the reference image, the minimum Pearson correlation desired, and the number of significant coefficients (N). The lowpass filter can be specified completely, or it can be chosen using one of the two lowpass filter design methods described above. The number of scales that the user wishes to test must also be specified (S). The default is $S = 6$.
2. The input and reference image are converted to polar coordinates.
3. The wavelet transform of both the input image and the reference image are taken at scales two through S .
4. The LH subband and the HL subband are extracted from the transformed data, and the significant coefficients at the first scale (scale two) are determined.
5. The coefficients are masked, where ones are placed in the locations of the significant coefficients, and zeros are placed elsewhere.
6. An initial estimate is obtained by comparing the 1st, 10th and middle significant coefficients of the LH subband of the reference image with all of the significant coefficient locations of the LH subband of the input image. This results in $3N$ initial estimates of rotation. The same operation is completed using the HL subband, producing another $3N$ estimates.

7. The LH subband of the reference image is shifted by each of the 3N estimates of the LH subband. The LH subband of the reference image and the LH subband of the input image are correlated for each of the rotation estimates for the LH subband. The rotation of the LH subband that has the best correlation is taken as the proper rotation for that subband. This step is then completed for the HL subband, which gives us the LH subband rotation estimate and the HL subband rotation estimate.
8. The reference image is rotated to each of the rotation estimations, and the Pearson correlation is computed using each of the rotated reference images and the input image. The estimate with the best correlation is taken as the correct rotation.
9. If the Pearson correlation value does not meet the user specified minimum Pearson correlation required, the algorithm will begin again at step three using the values at the next highest scale.
10. The input image is rotated to the reference image based on the solution to step eight once the minimum Pearson correlation requirement is met. If the minimum Pearson correlation requirement is not met, the highest correlation achieved will be chosen.

3.6.1 Cartesian Coordinates to Polar Coordinates. The first step to using the rotation algorithm is to convert the image in Cartesian coordinates to polar coordinates. The following equations were used to convert the image to polar coordinates:

$$\begin{aligned}
 r &= \sqrt{x^2 + y^2} \\
 \theta &= \begin{cases} \arctan \frac{y}{x} & \text{if } x > 0 \\ \pi + \arctan \frac{y}{x} & \text{if } x < 0 \end{cases}
 \end{aligned}$$

In this equation, the x and y are Cartesian coordinates, and the r and θ are polar coordinates [13].

Figure 3.5(a) shows an image in Cartesian coordinates, and Figure 3.5(b) shows the image after being converted to polar coordinates. Once this transformation is complete, the rows will correspond to the radii and the columns will correspond to the angles. Any sampling along the angles is possible, we choose integer angles from one to 360 degrees, so that a shift of one degree will correspond to a shift by one column.



Figure 3.5. *Converting Cartesian Coordinates to Polar Coordinates. (a) Camera-man image in Cartesian coordinates. (b) Cameraman image in polar coordinates.*

3.6.2 The Rotation-Invariant Transform. Our rotation-invariant transform is a two-dimensional transform. A lowpass filter was chosen by finding the best fit to the Daubechies 9,7 discrete wavelet $h(n)$. The continuous wavelet transform is the highpass filter. The rows in each image are analyzed first. For the rotation algorithm, the rows correspond to the radii and the columns correspond to the angles. Therefore, the LL subband would correspond to lowpass filtering along the radii and lowpass filtering along the angles of the resulting image. This is the coarse approximation of

the image as before. The LH subband corresponds to lowpass filtering along the radii and highpass filtering along the angles. This will preserve edges along the radii and has a blurring effect on the edges along the angles. The HL subband corresponds to highpass filtering along the radii and lowpass filtering along the angles of the resulting image. This subband preserves the edges that are along the angles and blurs the edges along the radii. The HH subband corresponds to highpass filtering along the radii and highpass filtering along the angles. This subband preserves the edges that persist across the radii and the angles together. Figure 3.6 shows scale two of the polar multiscale transform for the cameraman image.

3.7 Registration Algorithm Details

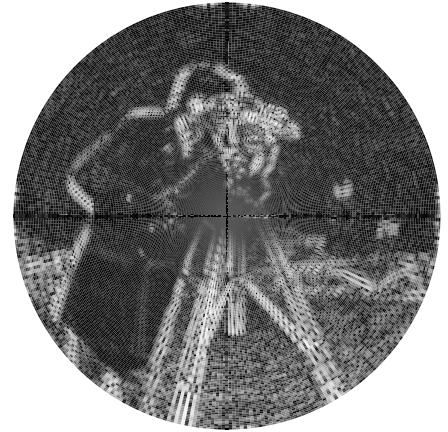
This section will cover the details of the translation and rotation registration algorithms that are common to both algorithms.

3.7.1 Subband Choice. It has been previously shown that the LH subband and the HL subband are most appropriate for image registration of translation and rotation changes. According to Stone et. al. [33], the HH subband is adversely affected by noise because noise is composed of high frequency components. Therefore, for images with large amounts of noise, the features (wavelet coefficients) that are extracted from the HH subband as significant coefficients may be inconsistent from image to image. The LL subband will also not work well in this algorithm because the coefficients are not sparse. In other words, it would take many more coefficients in the LL subband to adequately describe the image for registration than it would in any of the other three subbands.

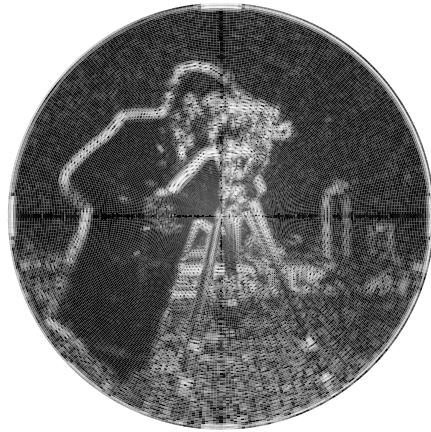
3.7.2 Choosing Significant Coefficients. There are two features unique to this work that are used in choosing significant coefficients. The first feature is used to account for problems with using the continuous wavelet transform. Since border effects exist within the subbands, they must be removed. The details of these border



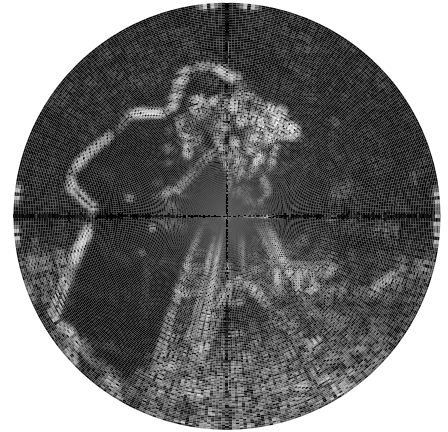
(a)



(b)



(c)



(d)

Figure 3.6. *Scale two of the Polar Multiscale Transform of the Cameraman Image. (a) The LL subband. (b) The LH subband. (c) The HL subband. (d) The HH subband. The radii correspond to the rows, and the angles correspond to the columns.*

effects are discussed in Section 2.5.1; the border effects are removed by putting zeroes in their place. These zeros take the form of a frame around each subband, which take the place of the ten rows on the top and bottom of each subband and the ten columns on the left and right of each subband. This was necessary so that these coefficients were not chosen as significant coefficients. The ten pixel width of the zero frame was analytically determined by viewing the multiscale transform of images at different scales.

Getting rid of the border effects of an image in Cartesian coordinates is simple because the four outer borders of the array correspond to the four outer borders of the image. See Figure 3.7. If the images are in polar coordinates, this is not the case. If the borders are framed as in the translation algorithm, the center of the image would be removed and a pie shape where the radii begin would be removed, as well as the outer ring of the image. This shape is similar to a “pacman” shape, see Figure 3.8. The center of the image could contain significant data, so we must determine another way to fix the border effect problem. Experimentation was done to find the appropriate method. Two methods were considered. The first method involved reflecting the first ten pixels in each row and putting this data in the front of the array, as well as reflecting the top 10 and bottom 10 pixels of each column in the array and adding them to the top and bottom of the array, respectively. The last ten pixels on each row will be zeroed out as before, since they correspond to the outer ring of the image. This larger image is then used as the input to the rest of the algorithm.

The second method involved using “real data,” in which the ten pixels from the center of the radius opposite the data in the image were put in front of the corresponding row. In this case, the first ten pixels of the middle row would be put in front of the first row. The columns and the pixels at the end of the rows were handled in the same manner as the first method. This image was then used as input for the rest of the algorithm. These two methods had similar registration accuracy,

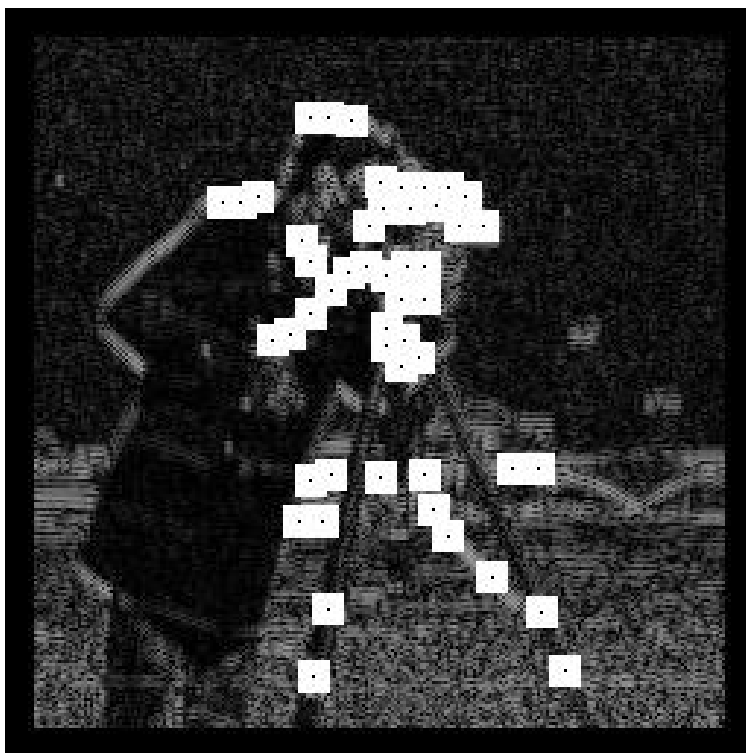


Figure 3.7. *Example of the “zero frame” and “black-out” regions. The black frame around the outside of the image was used to correct for errors due to border effects. The second piece of information that is important in this image is the “black-out” regions. In this image, we see an example of the method used to determine the N significant coefficients for this algorithm. The LH subband is shown here, with $N = 50$ significant coefficients chosen. The significant coefficient that was chosen is in the center of each square, shown as a black dot in the center of each white square. The “black-out” regions are the 11×11 white squares surrounding each significant coefficient.*

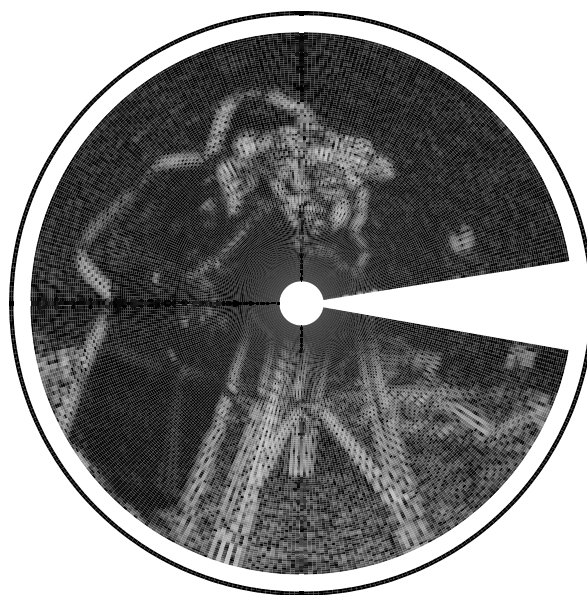


Figure 3.8. *Example of the “zero frame” region with no periodization for polar images. The white frame that was around the outside of the image in Cartesian coordinates takes on a “pacman” shape in polar coordinates. We correct this issue by periodization.*

so the first method was chosen for ease of computation. Figure 3.9 shows the frame around the outer border of the image in polar coordinates.

The second feature used in selecting significant coefficients is used to enforce spatial separation of the significant coefficients. In work by Brown [5], the N significant coefficients were chosen by taking the N largest magnitude coefficients. In our algorithm, we found that the significant coefficients were not adequately separated due to clustering. To address this issue, we developed a method which creates a “black-out” region around each chosen significant coefficient. This “black-out” region is an exclusion zone in which zeros are placed in an $R \times R$ square around the significant coefficient. See Figure 3.7. The pixel in the center of each of the white squares is the pixel that was chosen as the significant coefficient. No other significant coefficients can be chosen from an exclusion zone. This ensures that the significant coefficients are spread out. In the rotation algorithm, the “black-out” regions do not correspond to squares because the grid in polar coordinates is not uniform. Therefore, in Figure 3.9 we can see the odd shapes of the “black-out” regions, but within the array, these regions correspond to squares.

There are several methods to finding the significant coefficients using the “black-out” regions. The most obvious method is to sort the wavelet coefficients and select the largest coefficient in the list. Then, all of the coefficients that are in the $R \times R$ region surrounding that largest coefficient would be excluded. Finally, the wavelet coefficients would be resorted, and the largest coefficient in that list would be chosen. This process would be repeated until there were N coefficients, which would have to be stored in a separate matrix. This method is very time consuming because the sorting function is a very slow and computationally intensive function.

The algorithm that we developed involves finding the largest coefficient, then zeroing out the coefficients around that coefficient as well as the largest coefficient. A placeholder is placed in the location of the largest coefficient in another matrix of the same size. The largest coefficient is then found from the remaining coefficients

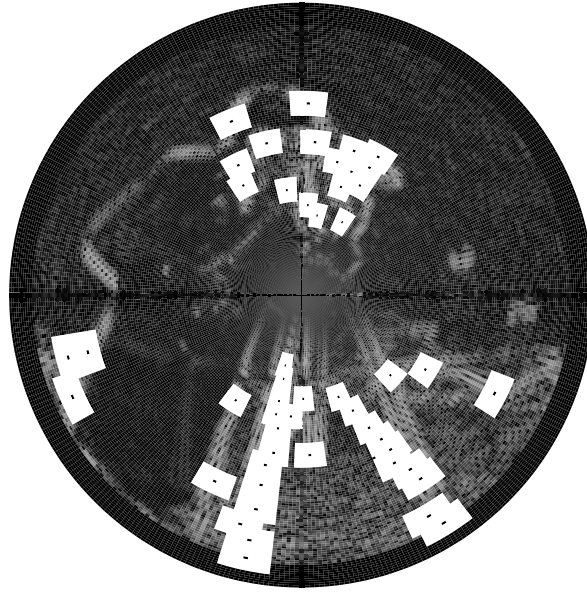


Figure 3.9. *Example of the “zero frame” and “black-out” regions for polar images. The black frame around the outside of the image was used to correct for errors due to border effects. The second piece of information that is important in this image is the “black-out” regions. In this image, we see an example of the method used to determine the N significant coefficients for this algorithm. The LH subband is shown here, with $N = 50$ significant coefficients chosen. The significant coefficient that was chosen is in the center of each square. The “black-out” regions are the 11×11 white squares surrounding each significant coefficient.*

in the original matrix. This process is repeated until there are N coefficients. This algorithm works significantly faster than the previous method because a function to find the maximum is all that is needed, instead of a computationally intensive sorting function. Our technique also removes several loops that are needed when using the sorting function for this purpose.

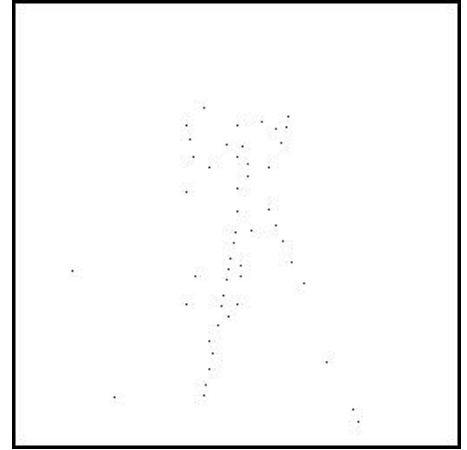
3.7.3 Masking. Once the N significant coefficients are found, they are masked. The masking method that was incorporated into this algorithm was originally created by Brown [5]. In the masking algorithm, an array of zeros the same size as the LH or HL subband is created. A value of one is placed in the location of each of the N significant coefficients instead of the actual value of that coefficient. This method reduces the noise effects and allows for a more efficient way to narrow down the $3N$ initial estimations of each subband to a final estimation for each subband. Figure 3.10 shows a representation of masking for the translation and rotation algorithms.

3.7.4 Initial Estimation. An initial estimate is obtained by comparing the 1st, 10th and middle significant coefficients of the LH subband of the reference image with all of the significant coefficient locations of the LH subband of the input image. Basically, the translation or rotation estimate for the 1st significant coefficient of the reference LH subband compared to the first significant coefficient of the input LH subband is the difference between the respective locations of these two coefficients in each subband. This results in $3N$ initial estimates of translation or rotation. The reference image is then shifted by each of the $3N$ estimates for the LH subband.

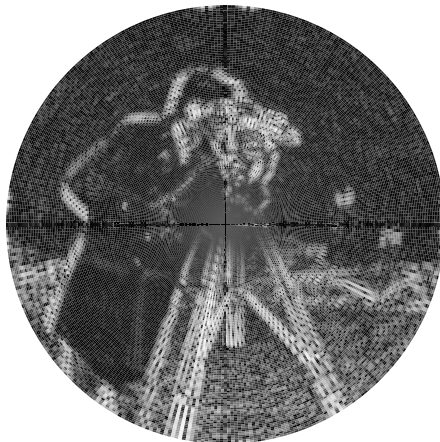
The LH subband of the reference image and the input image are then correlated. The correlation is performed by multiplying the two masked images, and then adding up the hits (places where there are ones in both images). Due to the masking, this is a purely binary operation. The translation or rotation for the LH subband that has the best correlation (the most hits) is taken as the translation



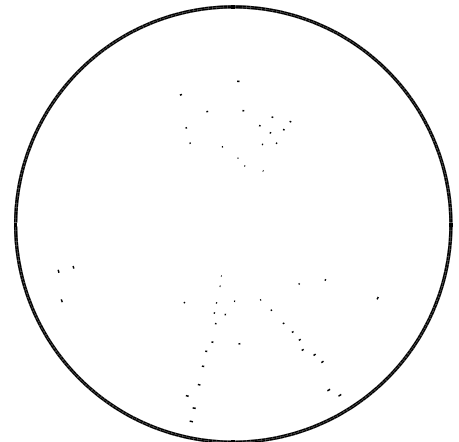
(a)



(b)



(c)



(d)

Figure 3.10. *Example of the masking algorithm. (a) The original subband before masking in Cartesian coordinates. (b) The subband after masking in Cartesian coordinates. (c) The original subband before masking in polar coordinates. (d) The subband after masking in polar coordinates. The black dots are the locations of the significant coefficients.*

or rotation estimate for the LH subband. Using the actual values of the significant coefficients would require a real-valued cross correlation. This is a much more time-consuming process than simply multiplying the two binary masks and adding up the hits. Therefore, the computation time is greatly reduced by using the mask.

Next, the same procedure is followed for the HL subband. The 1st, 10th, and middle significant coefficients of the HL subband of the reference image are compared with all of the N significant coefficients of the HL subband of the input image, which creates another $3N$ estimates. The reference image is shifted by each of the $3N$ estimates for the HL subband. The HL subband of the reference image and the input image are correlated, using the same method from above. The translation or rotation of the HL subband that has the best correlation is taken as the translation or rotation estimate for the HL subband.

3.7.5 Final Estimation. The initial estimation process produces two estimates of translation or rotation, one for the LH subband and one for the HL subband. The next step is to choose one of the two estimates as the final estimate of translation or rotation. The original reference image is shifted to the LH estimate of translation or rotation and correlated with the input image using the Pearson correlation equation. Then the reference image is shifted to the HL estimate of translation or rotation and correlated with the input image using the Pearson correlation equation. The Pearson correlation equation is shown below:

$$correlation = \frac{\sum_k (X(k) - \bar{X}) * (Y(k) - \bar{Y})}{((N - 1) * \sigma_X * \sigma_Y)} \quad (3.1)$$

where X is the reference image, Y is the input image, \bar{X} is the mean of the reference image, \bar{Y} is the mean of the input image, N is the number of significant coefficients, σ_X is the standard deviation of the reference image, and σ_Y is the standard deviation of the input image. The estimate corresponding to the subband with the best correlation is taken to be the final estimate of the translation or rotation.

3.7.6 The Multiscale Estimation. If the final estimation does not have a Pearson correlation greater than or equal to the user specified minimum Pearson correlation, then the next scale of the multiscale transform is used and the whole process is repeated. This approach allows the use of as many scales as necessary to achieve the required accuracy. If the minimum Pearson correlation is specified as one, all of the scales will be utilized to find the best estimate, since the Pearson correlation will not reach one with noisy images. If scale S is reached, and the minimum Pearson correlation value has not been achieved, the algorithm chooses the registration estimate of the scale that had the best correlation.

A study was done to find a value for the minimum Pearson correlation that yields the best trade-off between efficiency and accuracy. Figure 3.11 shows the results of this study. To obtain the results shown in Figure 3.11, four images were tested. The input image was shifted four pixels in the positive x direction and seven pixels in the negative y direction when compared to the reference image. The images that were tested include Lenna with 30 dB PSNR, Cameraman with 30 dB PSNR, Lenna with 22 dB PSNR, and Cameraman with 22 dB PSNR. Each of these images were tested 50 times at each minimum Pearson correlation value to obtain 200 total runs at each minimum Pearson correlation value. The percentage correct and the time for these runs were then averaged. Two hundred and fifty significant coefficients were kept in this experiment. This value was chosen because a high percentage correct was obtained at this value in other experimentation. Figure 3.11(a) shows the minimum Pearson correlation value versus the average time it took to complete one run of the multiscale transform registration algorithm. From these results, we see that it takes about the same amount of time for one run of the multiscale registration algorithm regardless of the minimum Pearson correlation that must be achieved. Therefore, we turn to the average percentage correct to determine which minimum Pearson correlation value to choose. From Figure 3.11(b), we see that a good percentage correct was obtained from a minimum Pearson correlation of 0.96. Therefore,

this value was chosen as the minimum Pearson correlation for use in later testing. In addition to the average percentage correct being high, the variance in both the x and y direction was low, and the means in the x and y directions were low.

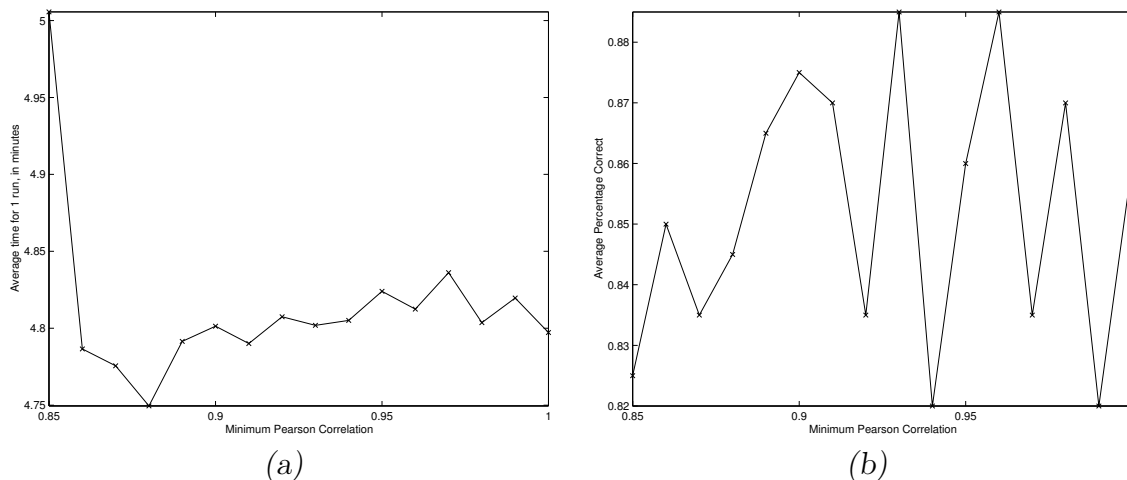


Figure 3.11. *Finding the appropriate value for the minimum Pearson correlation.*
 (a) *Minimum Pearson correlation versus time required for one run of the registration algorithm averaged over 50 runs of four images.*
 (b) *Minimum Pearson correlation versus average percentage correct averaged over 50 runs of four images.*

3.8 The Scale Algorithm

The scale algorithm follows similar steps to the translation and rotation algorithms. The same transform is used as in the other algorithms. However, the process is slightly different. The user again specifies the input image, the reference image, and the number of significant coefficients (N). The lowpass filter can be specified completely, or it can be chosen using one of the two lowpass filter design methods described above. The number of scales that the user wishes to test must also be specified (S). The default for this is $S = 6$. The multiscale transform of the input and reference images are taken at scales two through S . The LH subband and the HL subband are extracted from both the reference and the input image. The N significant coefficients are determined in the same manner as above for scales two through S of each subband for both images. The coefficients are masked, where ones

are placed in the locations of the significant coefficients, and zeros are placed elsewhere. The N coefficients of the reference image are compared to the N coefficients of the input image at each scale to create $S - 1$ estimates for both the LH subband and the HL subband.

At this point there is a deviation from the previously mentioned algorithms. Since the coefficients chosen are similar across scale, some significant coefficients match, even when the signals look largely different. See Figure 3.12. This figure represents the highpass portion of the multiscale transform. We show that the CWT of a signal and the CWT of a lowpass filtered version of the signal will eventually match at the scale corresponding to the width of the filter. For example, if the lowpass filtered signal was filtered using a quarter-band filter, then the signals should look similar at scale four. This figure contains two signals. The first signal (solid line) is a one-dimensional signal, and the second signal (x's) is the one-dimensional signal lowpass filtered by a quarter-band filter. We would expect the signals to be highly correlated at scale four because the response of a continuous wavelet transform at scale four should be similar to the response of taking the CWT of the signal lowpass filtered at this scale. The scales before scale four should not correlate because most of the energy in the lowpass signal has been removed at these scales (Figure 3.12 (b,c)). After the signals align, as in Figure 3.12(d), the remaining scales should continue to align because the signals are essentially the same at this point.

Figure 3.13 represents the lowpass portion of the multiscale transform. We show that a lowpass filter and a lowpass filter convolved with another lowpass filter will produce similar filters at the scale where the two lowpass filters have the same bandwidth. For example, if the original signal is lowpass filtered by a quarter-band filter, then the lowpass portion of the multiscale transform will begin to look similar at scale four. In Figure 3.13, we see that the two filters in (a) and (b) are largely different. At (c), the two filters begin to look similar, and in (d) and (e), the filters seem to match almost exactly.

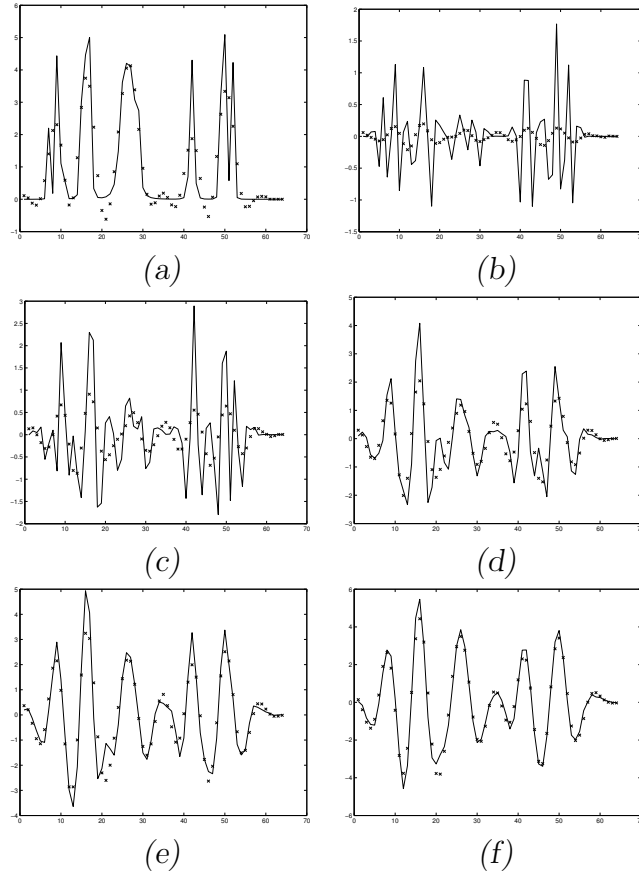


Figure 3.12. *Comparison of significant coefficients of a signal and a lowpass filtered version of the signal. In each of these figures, the solid line represents the original signal, and the x's represent the lowpass filtered signal. To create the lowpass filtered signal, a quarter-band lowpass filter was used. (a) The two signals before taking the CWT. (b) The signals at scale two of the CWT. (c) The signals at scale three of the CWT. (d) The signals at scale four of the CWT. (e) The signals at scale five of the CWT. (f) The signals at scale six of the CWT.*

Since the lowpass and highpass portions of a signal and a filtered signal align at the scale corresponding to the scale of the input signal when experimenting with the components of our multiscale transform, we should be able to determine the scale of an image by determining when the two signals align using the multiscale transform.

Unfortunately, some coefficients match below the correct scale. Because of this, there is a need for some sort of thresholding method, so that the appropriate scale can be chosen despite the fact that a small number of coefficients match at the incorrect scale. To perform this discrimination, two methods were designed.

The first method involves taking the minimum and maximum value of the significant coefficients of both images. The image with the lower range is considered to be the input image, and the image with the higher range is the reference image. All of the coefficients in the input image that are higher than the minimum value of the reference image are taken to be the significant coefficients for the input image. This means that the input image will have less than N coefficients to compare to the reference image, but the reference image will always have N coefficients. This will hopefully weed out the scales where little energy is present, and therefore, the right scale will be registered. The second method that was considered involved using the total energies at each scale. If the total energy of the input subband were less than some percentage of the total energy of the reference subband, then the images were considered to be at different scales.

The next step is to correlate the significant coefficients for the reference and the input subbands. The significant coefficients that were kept using the two methods above were correlated in the same manner as with the previously discussed algorithms. However, each scale of the reference subband is only correlated with the same scale of the input subband. This is because the reference image and the input image begin to match when taking the multiscale transform at the original scale of the input image, and continue to match after that scale. The $S - 1$ estimates for each subband are reduced to one estimate for each subband. The correct estimate

for each subband is considered to be the first subband that contained any hits, which corresponds to some correlation greater than zero. The Pearson correlation of the input image and the reference image at the appropriate scale are taken, and the subband with the greatest correlation is considered to be the correct estimate of the scale. Unfortunately, both methods provide sporadic results when registering images with scale differences. However, this algorithm shows promise and should be explored further.

3.9 Summary

In this chapter, we have developed a translation-invariant multiscale transform as well as a rotation-invariant multiscale transform which can register images across scales. There are several aspects of these algorithms that are new. First, these algorithms are multiscale algorithms. They utilize as many scales as necessary to get the job done to the degree of accuracy required. The second new aspect of these algorithms are that they utilize the two-dimensional continuous wavelet transform and a lowpass filter to obtain subbands similar to the redundant discrete wavelet transform. This transform increases the number of scales that can be utilized over the redundant discrete wavelet transform. The redundant discrete wavelet transform creates wavelet coefficients on dyadic scales only. Another aspect that is unique to these algorithms is the creation of an algorithm to spatially separate the significant coefficients by an amount specified by the user.

In the next chapter, we will discuss the performance of the image registration algorithms. Determining the scale of an input image relative to a reference image will not be demonstrated due to its sporadic nature. However, we determined that our translation and rotation registration algorithms are invariant to scale changes. Since translations and rotations can be determined across scale, we do not need to remove the scale to determine the translations and rotations.

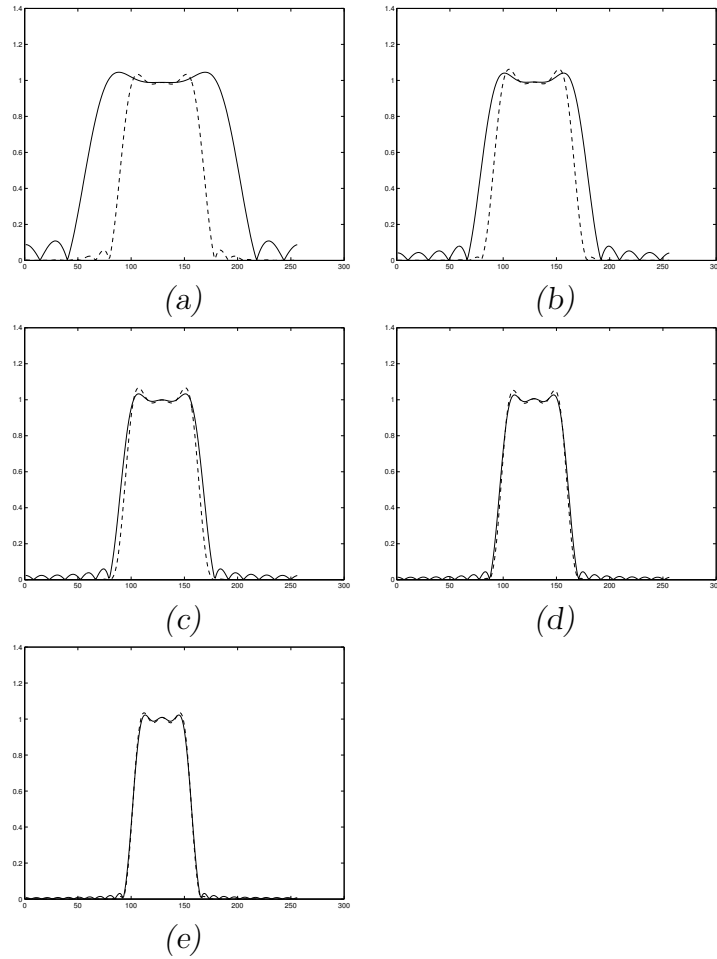


Figure 3.13. *The alignment of the lowpass filters when the scales align. The input images are created by lowpass filtering the original image. In this figure, the filter is a quarter-band lowpass filter. The solid line shows the lowpass filter that is used in the multiscale transform, and the dashed line shows the quarter-band lowpass filter convolved with the lowpass filter at the given scale. (a) shows the frequency responses of the half-band lowpass filter and the quarter-band lowpass filter convolved with the same half-band filter. (b) shows the frequency response of the third-band lowpass filter and the quarter-band filter convolved with the third-band lowpass filter. (c) shows the frequency response of the quarter-band lowpass filter and the quarter-band filter convolved with the quarter-band filter. (d) shows the frequency response of the fifth-band filter and the fifth-band filter convolved with the quarter-band filter. (e) shows the frequency response of the sixth-band filter and the sixth-band filter convolved with the quarter-band filter. This figure shows that when the scale of the input image is four, the filters will begin to line up when both filters correspond to a quarter-band lowpass filter, and from then on, the filters align.*

IV. Results

4.1 Introduction

In this chapter, we will demonstrate the effectiveness of our algorithm with validation studies. We begin with a discussion on the design of the validation studies, then we present the results of these studies. We perform four studies for both the rotation and translation algorithms. These studies can be separated into two categories.

The first category contains those studies which involve registering images at the same scale. The first study in this category compares the algorithm created by Brown [5] with a single scale of our new multiscale registration algorithm. Next, we will look at multiscale results, where an input image will be registered to a reference image at scales two through six of the multiscale transform. The registration accuracy at each of these five scales will be shown. In addition, the registration results of scale four of our algorithm will be compared to the equivalent scale two results using Brown's algorithm. The scale two registration of the Brown [5] algorithm was not part of his experimentation. It has been added to our studies for comparison purposes. The final study in the category of same scale registration involves showing the accuracy of the registration algorithms described in Sections 3.5 and 3.6.

The next category of the validation studies includes those tests which register an input image to a reference image when the input image and reference image are at different scales. There is one validation study contained in this category. This study uses the registration algorithms in Sections 3.5 and 3.6 to register an input image at scale three to a reference image at scale zero. The translation validation studies will be presented first, followed by the rotation validation studies.

4.2 Design of the Algorithm Validation Studies

The objective of this research is to create an accurate, robust, and efficient registration algorithm. The studies that are shown below seek to show the performance of our new multiscale registration algorithm. In the following studies, an input image (translated or rotated version of a reference image) will be registered to a reference image to determine if our multiscale algorithm can accurately predict the translation or rotation differences that exist between the two images. With these studies, we hope to provide information on two other questions of importance to registration. First, we would like to be able to determine a number or range of coefficients that should be kept in order to provide the most accurate registration results using our algorithm. Secondly, we seek to determine when a multiscale registration algorithm would be preferable to a single scale registration algorithm. To answer these questions, we perform several studies. The design of these studies will be discussed in the following sections.

4.2.1 Test Images. Two images were selected to use in the validation studies. The two images are shown in Figure 4.1. The image in Figure 4.1(a) will be referred to as “Cameraman” in future discussions, and Figure 4.1(b) as “Lenna”. These images are 256×256 pixel grayscale images. The images were selected for two reasons. First, these images were used in Brown’s research [5]. Comparing the two algorithms will be easier because the images are the same. Secondly, the two images contain a central figure with nice edges, as well as regions in the background of low contrast, which makes them good test images.

When registering an input image to a reference image where both images are at the same scale, the images in Figure 4.1 will be considered the reference images. Shifted or rotated versions of these two images will be used as the input images.

When registering images across scale, the reference image is at scale zero. The multiscale transform of the reference image is needed to form the input image. The

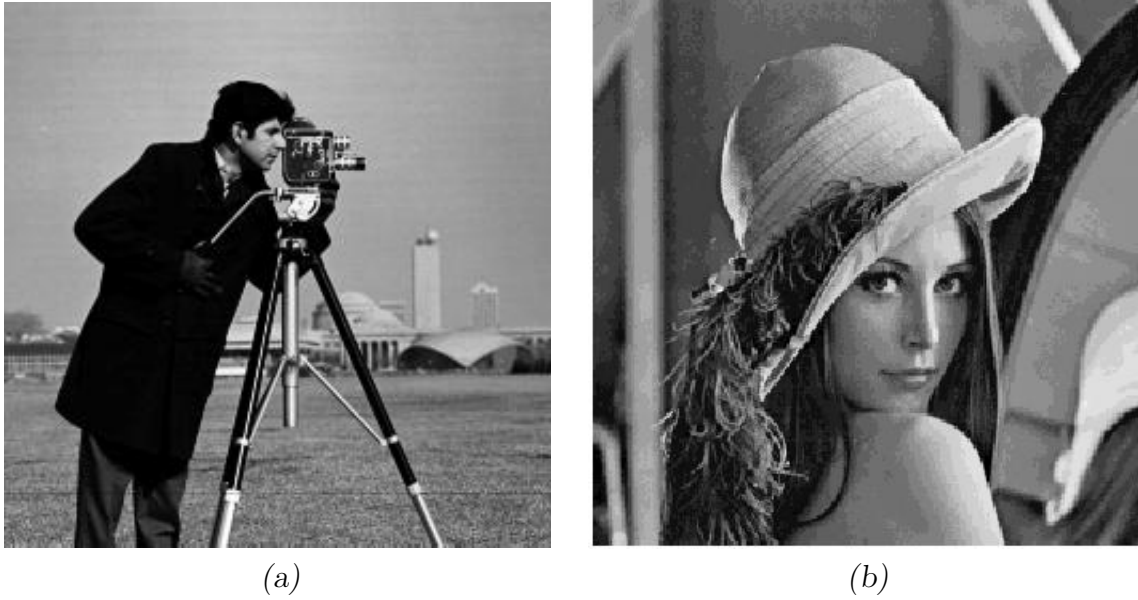


Figure 4.1. *Cameraman and Lenna images. (a) Cameraman image. (b) Lenna image. These images will be considered the reference images for testing the multiscale algorithm. The input images will be shifted or rotated versions of these images.*

LL subband is extracted from the multiscale transform at scale three and used as the input image. Therefore, a third-band lowpass filter is used to create the input image. Figure 4.2(a) shows the original Cameraman image (scale zero). This image will be used as a reference image. Figure 4.2(b) shows the LL subband of the multiscale transform at scale three for the Cameraman image. A shifted or rotated version of this image will be the input image when Cameraman is the reference image for testing the registration algorithm at different scales. Figure 4.2(c) shows the original Lenna image (scale zero), and Figure 4.2(d) shows the LL subband of the multiscale transform at scale three for the Lenna image. The Lenna image will be the reference image, and a shifted or rotated version of Figure 4.2(d) will be the input image to the registration algorithm for different scale registration.

4.2.2 Peak Signal-to-Noise Ratio. To simulate real conditions, white Gaussian noise was added to the test images. In the case where the test images were at the



(a)



(b)



(c)



(d)

Figure 4.2. *Images used to test our multiscale registration algorithm when the reference and input images are at different scales. (a) Camera image. This image is the reference image. (b) The LL subband of Cameraman at scale three of the multiscale transform. A shifted or rotated version of this image will be considered the input image in later studies. (c) Lenna image. This image is the reference image. (d) The LL subband of Lenna at scale three of the multiscale transform. A shifted or rotated version of this image will be considered the input image in later studies.*

same scale, noise was added directly to the images. In the case where the test images were at different scales, noise was added to the input image after the LL subband at scale three was extracted, and noise was added directly to the reference image. The level of noise added to the images was measured using the Peak Signal-to-Noise Ratio (PSNR). The following equation describes the PSNR value:

$$PSNR = 20 \log \left(\frac{\max |x_i|}{\sqrt{\sum_i \frac{(x_i - \hat{x}_i)^2}{N}}} \right). \quad (4.1)$$

The x_i 's in Equation 4.1 are the grayscale values of the pixels in the original image (image before noise is added), and the \hat{x}_i in Equation 4.1 are the grayscale values of the pixels in the image where noise has been added. N refers to the number of pixels in the original image.

In the validation studies, two noise concentrations were tested. These noise amounts were consistent with the noise amounts chosen by Brown [5], so that the two studies could be easily compared. The PSNR values that were chosen are 30 dB and 22 dB. A 30 dB PSNR value is approximately the lower limit on noise that would be considered insignificant to the human eye. Any value higher than 30 dB PSNR is considered uncorrupted. When we have noise values that make the PSNR fall below 30 dB, the eye can detect that the image contains noise. A PSNR of 22 dB represents a significant degradation by noise. Figures 4.3(a) and (b) show the test images with a PSNR of 30 dB, and Figures 4.3(c) and (d) show the test images with PSNR values of 22 dB.

4.2.3 The Validation Study. The registration algorithm was tested by comparing a reference image to an input image. In the first case, translation registration using scale two of the new multiscale transform is compared to Brown's registration algorithm at equivalent scale one. Second, the results are shown for translation registration using scales two through six of the multiscale transform. In this study,



(a)



(b)



(c)



(d)

Figure 4.3. *Examples of PSNR values for the test images. (a) The Cameraman image with 30 dB PSNR. (b) The Lenna image with 30 dB PSNR. (c) The Cameraman image with 22 dB PSNR. (d) The Lenna image with 22 dB PSNR.*

registration using scale four of the multiscale transform is compared to the translation algorithm developed by Brown at scale two. Third, the multiscale algorithm is tested by using a minimum Pearson correlation value to obtain one translation estimation of the algorithm in the least amount of time, as described in Section 3.5. Finally, the multiscale algorithm in Section 3.5 is used to determine the translation when the reference image and the input image are at different scales. In our case, the reference image will be the Cameraman or Lenna image, and the input image will be a shifted version of the LL subband of the reference image at scale three of the multiscale transform. The same four validation studies will be shown for the rotation registration algorithm.

The following test images were used to determine the accuracy of the translation registration algorithm. The input image was shifted by four pixels in the positive x direction and seven pixels in the negative y direction. The translations are performed by circularly translating the input image by the given amount. Figure 4.4(a) shows the original Cameraman image, and Figure 4.4(b) shows the Cameraman image translated by four in the positive x direction and seven in the negative y direction. Figure 4.4(c) shows the original Lenna image, and Figure 4.4(d) shows the Lenna image translated by four in the positive x direction and seven in the negative y direction.

For the rotation algorithm, the image is first converted to polar coordinates, as discussed in Section 3.6.1 and shown in Figure 3.5. In polar coordinates, a rotation of one degree corresponds to a circular shift along the columns. The input image was rotated by 13 degrees to test the accuracy of the rotation registration algorithm. Figure 4.5(a) shows the original Cameraman image, and Figure 4.5(b) shows the Cameraman image rotated by 13 degrees. Figure 4.5(c) shows the original Lenna image, and Figure 4.5(d) shows the Lenna image rotated by 13 degrees. All of these images are in polar coordinates.



(a)



(b)



(c)



(d)

Figure 4.4. *Translated versions of the test images. (a) Cameraman image in Cartesian coordinates. (b) Translated version of Cameraman, where the image is shifted by four pixels in the positive x direction and seven pixels in the negative y direction. (c) Lenna image in Cartesian coordinates. (d) Translated version of Lenna, where the image is translated by four pixels in the positive x direction and 7 pixels in the negative y direction.*



(a)



(b)



(c)



(d)

Figure 4.5. *Rotated versions of the test images. (a) Cameraman image in Polar coordinates. (b) Rotated version of the Cameraman image, where the image is rotated by 13 degrees. (c) Lenna image in Polar coordinates. (d) Rotated version of the Lenna image, where the image is rotated by 13 degrees.*

In each of the studies, the reference images were translated or rotated by the appropriate value to create the input image. Noise was added to the images to obtain PSNR values of 30 dB and 22 dB. The number of significant coefficients was varied from 10 to 490 in steps of 20 for both the translation and rotation registration algorithms. To achieve statistically valid results of the performance of our algorithm, 50 realizations of noise were averaged for each number of significant coefficients.

A correct prediction of the translation or rotation was considered to be one in which the exact value of the translation or rotation was determined. In this case, a translation value of four in the positive x direction and seven in the negative y direction would be the only translation that would be considered correct. A rotation of 13 degrees, in our case, would be the only rotation that would be counted as correct. In previous work by Brown [5], a correct prediction included values within ± 2 of the actual value, since the discrete wavelet transform cannot predict values within this range. All of the results shown in the following sections will show exact values only as correct predictions, and therefore the results may vary slightly from the original results presented by Brown [5].

4.3 Translation Performance

The four studies mentioned previously in Section 4.1 will be considered here separately. In the first section, we will consider those registration studies where the input image and the reference image are at the same scale. In the second section, we will discuss the results for the registration validation study in which the reference image is at a different scale compared to the input image.

4.3.1 Performance Results for Same Scale Registration. Three validation studies will be considered in this section. First, we will compare the scale two registration using the multiscale transform with a single iteration of Brown's algorithm [5]. Next, we will look at multiple scale translation registration results. The registra-

tion accuracy results of scales two through six using the multiscale transform will be shown. In addition, the second iteration of Brown’s algorithm [5] will be compared to scale four of the multiscale transform. Finally, the minimum Pearson Correlation value (determined in Section 3.7.6 as 0.96) will be used in the algorithm described in Section 3.5, and the results will be presented.

4.3.1.1 Single Scale Translation Performance Comparison. The first study involved determining if the results of scale two of the new multiscale algorithm performed similarly to a single iteration of Brown’s algorithm [5]. The results are shown as the number of coefficients kept as significant versus the percentage correct. Figure 4.6 shows the results of this study. Figure 4.6(a) shows the results for the Cameraman image with a 30 dB PSNR. Figure 4.6(b) shows the results for the Lenna image with a 30 dB PSNR. Figure 4.6(c) shows the results for the Cameraman image with a 22 dB PSNR, and Figure 4.6(d) shows the results for the Lenna image with a 22 dB PSNR.

The results from our new multiscale algorithm compare favorably to the results of Brown’s algorithm. Overall, both algorithms seem to give the same performance. For the 30 dB PSNR cases, Figures 4.6(a) and (b), both algorithms essentially have a percentage correct of 100%. With a PSNR value of 22 dB, Figures 4.6(c) and (d), there is more variation in the accuracy of the two algorithms. However, for the Cameraman image, Figure 4.6(c), the accuracy of our algorithm never falls below 90%. There seems to be a downward trend to this graph, meaning that as more coefficients are used to estimate the translation, the worse the percentage correct. For the Lenna image with a PSNR of 22 dB, Figure 4.6(d), there is low accuracy when a small number of coefficients are used. The graph has an upward trend, so the accuracy is getting better as the number of coefficients increases. This is what we would expect. With a small number of coefficients the accuracy should be low because there are not enough coefficients to adequately describe the image and distinguish it from another image to determine the translation. As the number of

coefficients increase, the image will have an increasingly greater number of points to describe it, which makes it more unique compared to a different translation of the image. At some point, there will be too many coefficients describing the image. This problem becomes worse when large amounts of noise are added to the images because coefficients that would not be significant become significant due to the noise. This will make a coefficient match to a coefficient in a translated version of the image, when in fact, it should not be used for matching at all.

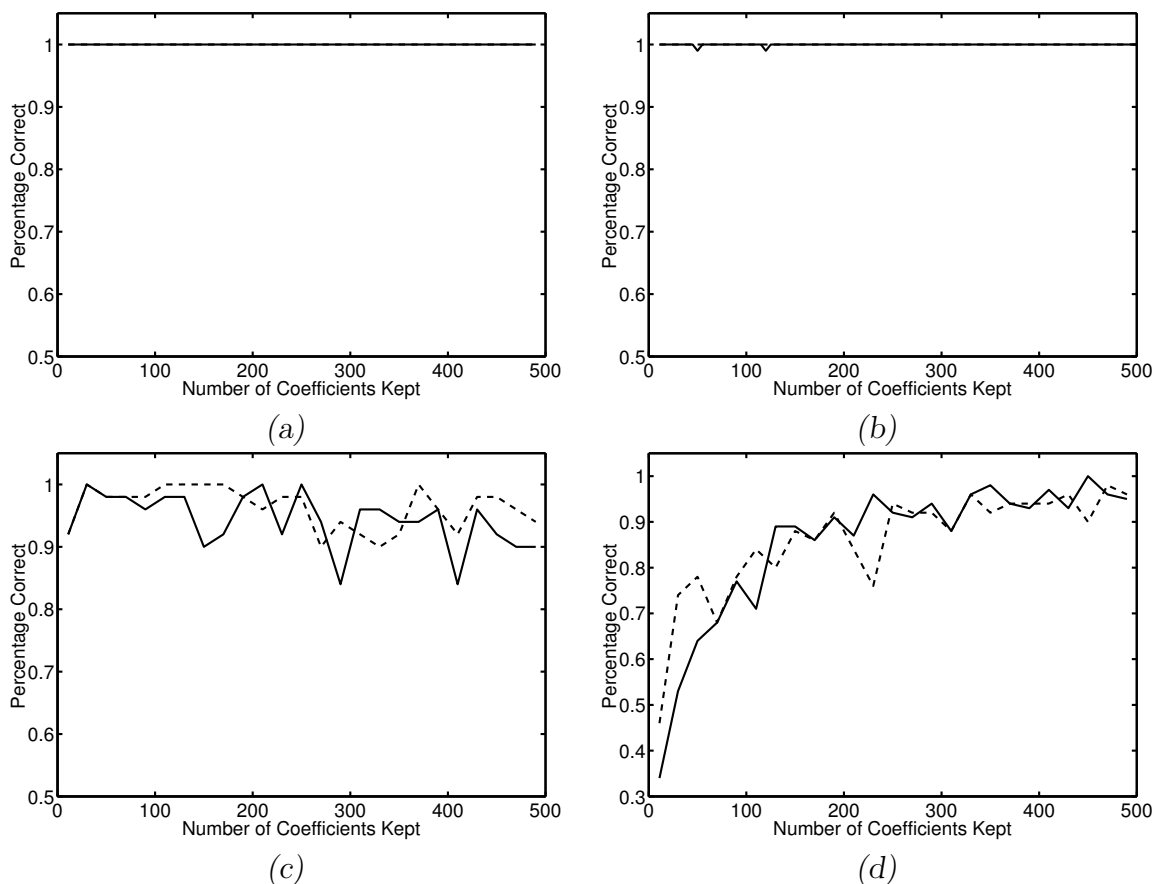


Figure 4.6. Results for scale two of the multiscale translation algorithm compared to the results of a single iteration of the algorithm developed by Brown [5]. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR. In (a)-(d), the solid line shows the results for the Brown algorithm, and the dashed line shows the results for multiscale translation registration.

4.3.1.2 Multiple Scale Translation Performance Comparison. The

following study determines the registration accuracy of scales two through six using the multiscale transform. This study will also compare scale four registration using the multiscale transform to the second iteration of Brown's algorithm. It is important to note that the algorithm developed by Brown can only produce results at dyadic scales, and therefore only scale four registration using our multiscale transform can be compared to the equivalent scale two of Brown's algorithm. In addition, Brown's algorithm, in its original form, registered images using the first iteration of the redundant DWT. His work has been extended here to produce single scale registration using the second iteration of the redundant DWT.

As scale is increased, the effect of noise is reduced because both subbands used for registration have been lowpass filtered. This is because the noise is white, and lowpass filtering preserves most of our signal energy. Therefore, we would expect registration accuracy to increase at higher scales. The registration accuracy at each of the five scales used in the registration algorithm is shown in Figure 4.7. Figures 4.7(a) and (b) show the registration accuracy for the two images with PSNR values of 30 dB. Since the accuracy at scale two is so high, little insight can be gained with respect to the registration performance of the different scales from looking at these two images. However, in Figure 4.7(b), we see that at scale two (the solid line) there are a few points where the accuracy is not 100% at the lower numbers of coefficients kept. Also, at scale three (the dotted line), there are a few points where the accuracy is less than 100%. But, scales four through six have an accuracy of 100% at all levels of coefficients kept. This is what we expect, because the accuracy increases as the scale increases. The accuracy cannot increase above 100%, so the accuracy cannot improve at higher scales once it has reached 100% accuracy.

When the images have a PSNR value of 22 dB, Figures 4.7(c) and (d), more insight can be drawn from the graphs. For the Cameraman image, Figure 4.7(c), the graph for scale two lingers around 100%, but has points where the percentage

correct is around 90%. There is a definite trend of increased accuracy with increased scale. This trend is very prevalent in Figure 4.7(d) because the accuracy at scale two was less than in the Cameraman image. Also, the trend is prevalent at the smaller numbers of coefficients, where each scale has a higher accuracy than the previous scale. At all scales, the percentage correct is low at small numbers of coefficients kept, and increases as the number of coefficients increases. By scale four, the accuracy of the algorithm is above 90% at all numbers of coefficients kept. The main point to draw from these graphs is that accuracy increases at higher scales, so we exploit this fact in our multiscale algorithm without greatly increasing the amount of time the algorithm takes to run over a single scale algorithm. Our multiscale algorithm uses these scales to improve registration accuracy, while the Brown algorithm is limited to a single scale. At most, if the Brown algorithm were converted to a multiscale algorithm, it can only use dyadic scales for registration, whereas our algorithm is able to exploit the scales between these scales to improve registration accuracy.

The results presented by Brown [5] used a single iteration (scale one) of the redundant DWT for image registration. We extend these results by performing registration using the second iteration of the Brown algorithm (scale two). Figure 4.8 shows the results of the comparison between the second iteration of Brown's algorithm and scale four registration using the multiscale transform. Scale four of the multiscale transform corresponds to the second iteration of Brown's algorithm, since the algorithm developed by Brown produces results only at dyadic scales. Figures 4.8(a) and (b) show the results for the images with a 30 dB PSNR. Since the percentage correct at scale two of both of these algorithms was essentially 100%, the two graphs for the 30 dB PSNR cases in Figure 4.8 look largely the same as Figures 4.6(a) and (b). However, in Figures 4.8(c) and (d), where the PSNR value is 22 dB, we can see that the accuracy of both algorithms has increased greatly. The accuracy of both algorithms never falls below 80%, where the scale two accuracy was as low as 30% with these algorithms. At scale four, these two algorithms obtain and

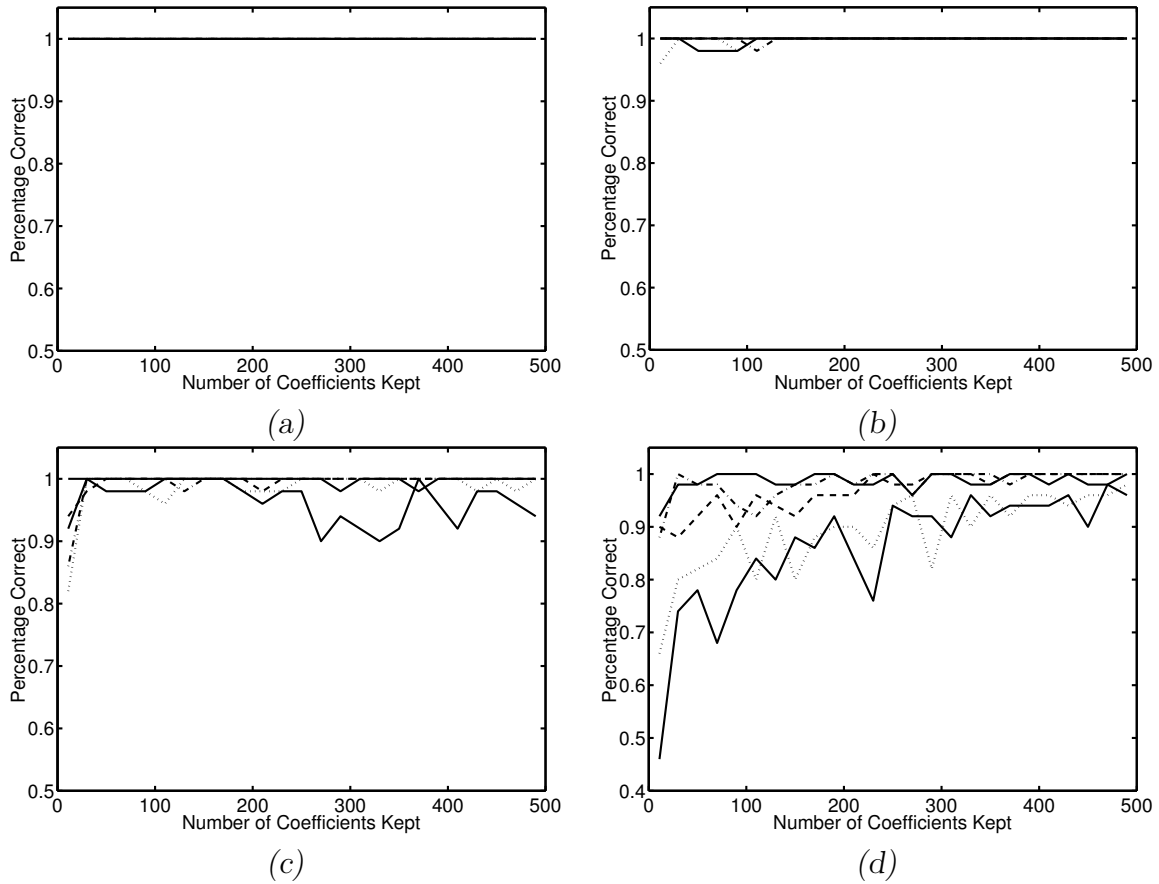


Figure 4.7. Results for translation registration using scales two through six of the multiscale transform. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR. In (a)-(d), the lower solid line shows scale two registration of the multiscale transform, the dotted line shows the scale three results, the dashed line shows the scale four results, the dot-dashed line shows the scale five results, and the scale six results are shown by a solid line as well. The scale six results are typically around one.

maintain 100% accuracy at a much quicker rate. These graphs show that using a multiscale approach with either algorithm will improve the accuracy of registration. Figure 4.8 shows that our algorithm performs equally to Brown’s algorithm in the multiscale sense.

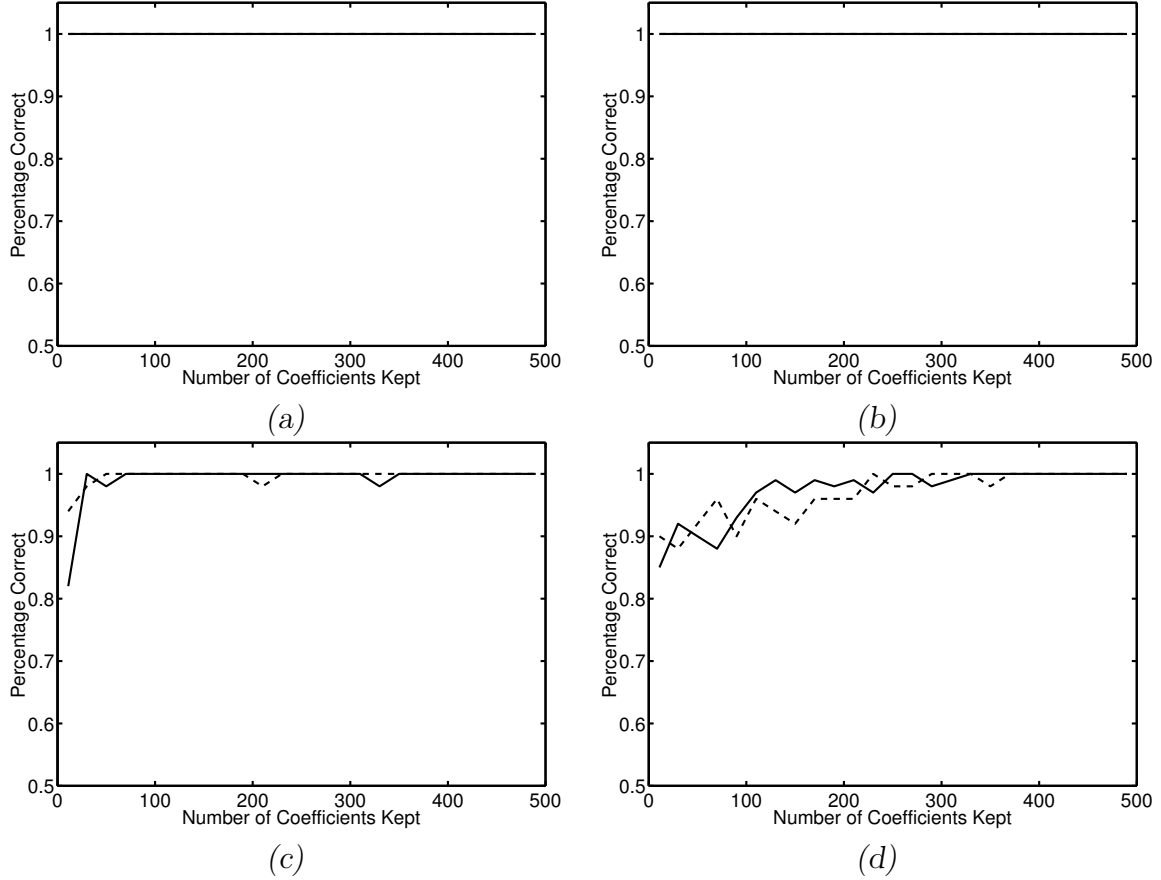


Figure 4.8. Results for scale four of translation registration using the multiscale transform compared to the results of the second iteration of the algorithm developed by Brown [5]. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR. In (a)-(d), the solid line shows the results for the second iteration of the Brown algorithm, and the dashed line shows the results for scale four registration using the multiscale transform.

4.3.1.3 Translation Performance for Same Scale Test Images. The

purpose of this study was to determine the accuracy of the algorithm described in

Section 3.5. Registration estimates could be computed for all scales, then we could choose the translation estimate of the scale with the best accuracy. However, this is time-consuming. By setting a minimum Pearson correlation value, we can obtain accurate registration by stopping at a scale that achieves this minimum value. This method will save us time over computing registration estimates at all scales. The minimum Pearson correlation value of 0.96 used in this algorithm was determined in Section 3.7.6. There is a trade-off between accuracy and time with the multiscale registration algorithm. To improve accuracy, the algorithm must spend more time to determine the translations that are needed because a greater scale must be used. The multiscale algorithm that we have developed seeks to find an adequate trade-off where we can obtain the most accuracy with the least amount of time spent. Therefore, two parts are needed to determine this. The first aspect of this study is to determine if this algorithm provides accurate registration, and the second aspect is to determine if the algorithm will work quickly. In Figure 4.9, we show the performance of the multiscale algorithm. The multiscale algorithm works well for registration of images that are translated compared to a reference image, as shown in Figure 4.9. The registration accuracy was 100% in all cases of coefficients kept.

Now that we have shown that the algorithm has good registration accuracy, we must determine an appropriate number of coefficients to keep. The quicker we wish the algorithm to be, the lower the number of coefficients we should use. However, we want to maintain a minimum amount of accuracy. Even though the accuracy is high in the test cases for low numbers of coefficients, a different image may not have such good performance, since the registration accuracy is somewhat image dependent. To ensure a good accuracy, a larger number of coefficients should be used. The best choice is between 100 and 250 coefficients because this range provides an acceptable trade-off between registration accuracy and computability. Clearly, there is a point where such a large number of coefficients is kept that noise begins to be a factor. The noise will make a less significant coefficient larger, causing it to be chosen as

significant, when in fact it is not. Therefore, the coefficients used to describe the reference image will not be the same as the coefficients used to describe the input image. In other words, the algorithm will be selecting different features from the two images, or worse yet, misinterpreting noise as image features.

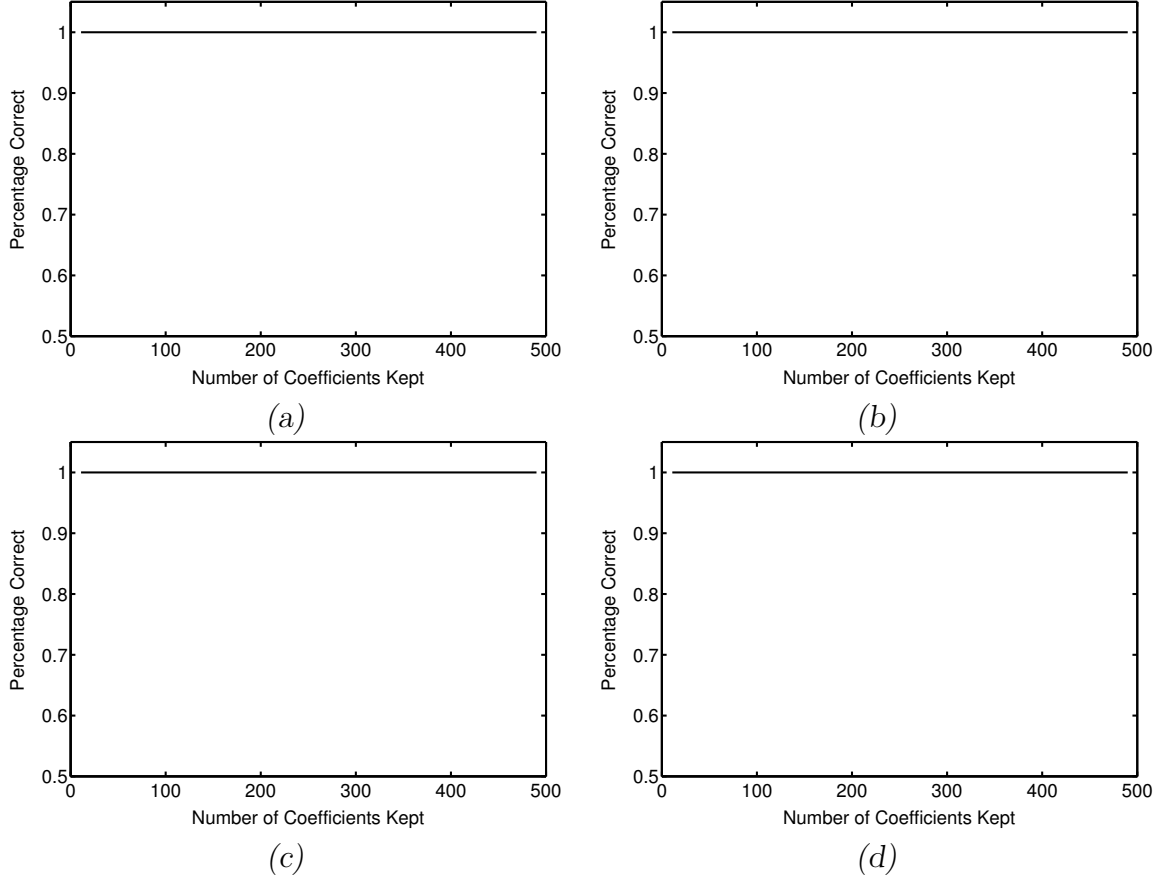


Figure 4.9. Results for the multiscale registration algorithm for translation where the input and reference images are at the same scale. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR. In (a)-(d), the solid line shows the results for the Brown algorithm, and the dashed line shows the results for the multiscale registration algorithm presented in 3.5.

The second aspect of this algorithm is to determine if the multiscale algorithm can run quickly by utilizing only the scales it needs to determine the accuracy to the minimum Pearson correlation value specified. This is difficult to show just by

using time comparisons because several different computers were used with varying speeds to produce the registration data. Therefore, we will show the efficiency of our algorithm by showing the scale that is needed to register an image versus the number of coefficients kept. The greater the scale, the more time the algorithm takes. If the average scale is around two, then the algorithm is about as quick as a single scale algorithm, yet equally or more accurate (as shown in Figure 4.9). The results for this study are shown in Figure 4.10. Using the translation registration algorithm, the registration accuracy of the two images was better than the single scale algorithm.

In the 30 dB PSNR cases, the average scale used was around two for both the Cameraman image and the Lenna image. This means that the multiscale algorithm will perform about as quickly as a single scale algorithm using scale two, with better accuracy, since the scale required to register some images is above scale two. If we choose coefficients in the range of 100 to 250, as mentioned earlier, then we can say that this algorithm performs better than the single scale algorithm in about the same amount of time for both the Lenna and the Cameraman images.

For the 22 dB PSNR cases, scale six was needed to register the test images. This means that the minimum Pearson correlation value was probably not met in most cases, and the best correlation was chosen, meaning all six scales were computed. The minimum Pearson correlation value should probably be adjusted, so that a lower scale can be used in registration. A lower minimum Pearson correlation value is needed to efficiently achieve good registration accuracy for larger noise cases. The larger the amount of noise in an image, the lower the correlation will be for a reference image and an input image translated to the reference image. However, since we are looking for an algorithm that will perform well on all images, independent of the amount of noise, the 0.96 minimum Pearson correlation value is an adequate number to obtain both good accuracy and computational efficiency. Using this value, a larger scale must be used to register the high noise cases, but the accuracy will be good for all noise cases. If information is known about the amount of noise in the images to be

registered, it might be beneficial to adjust the minimum Pearson correlation value, that is, make the value adaptive to an estimate of the image noise.

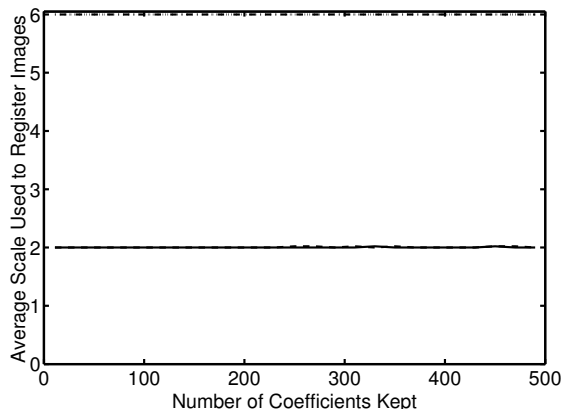


Figure 4.10. *Average scale used for each test image to register translation differences between two images when the images are at the same scale. The solid line shows the results for the Cameraman image with a 30 dB PSNR. The dotted line shows the Cameraman image with a 22 dB PSNR. The dashed line shows the Lenna image with a 30 dB PSNR, and the dash-dotted line shows the Lenna image with a 22 dB PSNR.*

4.3.2 Performance Results for Different Scale Registration. The purpose of this study was to determine the accuracy of the algorithm described in Section 3.5, where the input image is at a different scale compared to the reference image. The same method as described in Section 4.3.1.3 is used; a minimum Pearson correlation value of 0.96 is again used. There is a trade-off between accuracy and time with the registration algorithm. Therefore, we will show the same information as in the previous section. The first part of the study shows the algorithm's accuracy as the number of significant coefficients kept versus the percentage correct at that coefficient value. The second part of this study presents the efficiency of the algorithm, by showing the average number of scales necessary to determine the correct registration.

The results for the first part of the study are shown in Figure 4.11. We see the same trend as in the previous section for the four graphs in Figure 4.11. The algorithm performs well for both noise cases, with all of the images having approxi-

mately 100% registration accuracy. The 22 dB PSNR Lenna image, Figure 4.11(d), is the only image that does not have exactly 100% accuracy for all numbers of coefficients. This image illustrates the point of the importance of choosing an appropriate number of coefficients to keep. If we choose 10 coefficients, our accuracy would be 92%. By increasing the number of coefficients to 30, our accuracy increases to 100%. Therefore, because the registration accuracy is somewhat image dependent, we should keep between 100 to 250 coefficients to ensure that we are not in the range where the algorithm has not reached its full potential. Also, by staying in this range, we do not get to the point of noise corrupted coefficients, which is not visible in this figure.

The results for the second part of this study are shown in Figure 4.12. These results show the efficiency of the multiscale registration algorithm. In the 30 dB PSNR cases, the average scale used is between three and four. Therefore, a single scale algorithm at scale four would need to be used to obtain a similar accuracy as the multiscale algorithm. However, since some of the images would require a higher scale to register correctly, the multiscale algorithm provides better accuracy overall. For the 22 dB PSNR test images, scale six was needed to register the images. In the multiscale algorithm, a scale six value means that the images were registered at scale six or the algorithm went through scale six and did not obtain the minimum Pearson correlation required, and the best Pearson correlation achieved was kept. By adjusting the minimum Pearson correlation value, this scale should go down. However, since we are looking for an algorithm that will register images at any noise value, the 0.96 value is a good value to use, but more noisy images will require larger scales to register accurately.

From this study, we conclude that the multiscale algorithm provides more accurate registration when compared to a single scale of the Brown algorithm because this algorithm allows us to utilize multiple scales of the transform. This algorithm also allows us to obtain high accuracy in a relatively short amount of time, by

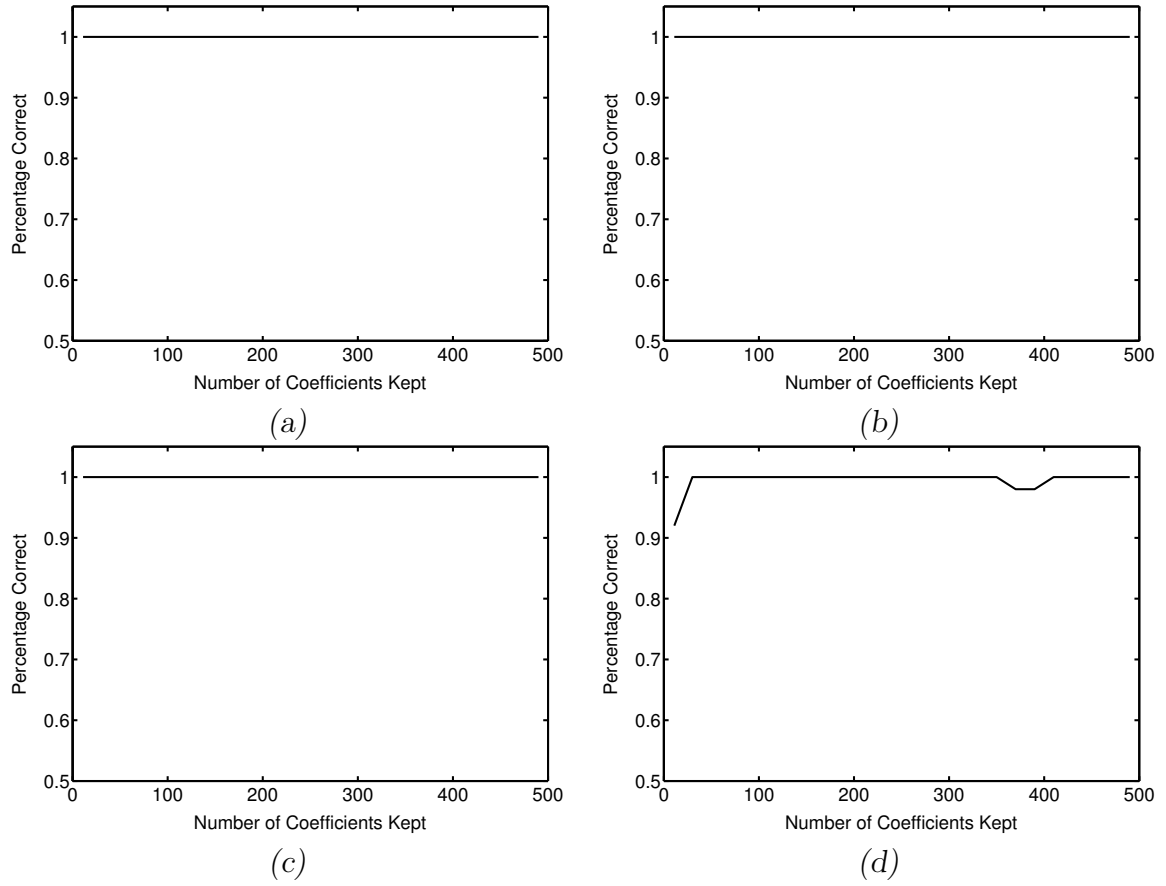


Figure 4.11. Results for the multiscale registration algorithm for input and reference images at different scales. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR.

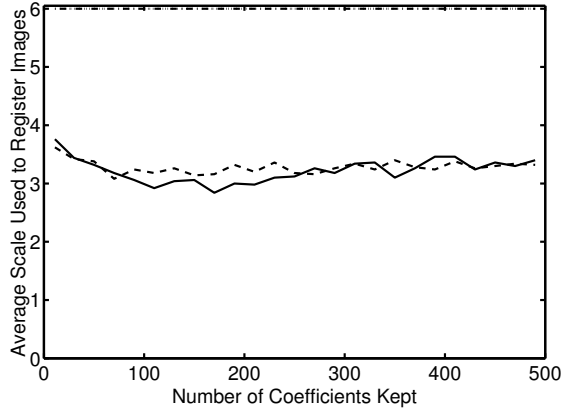


Figure 4.12. Average scale used for each test image to register the two images when the images are at the different scales. The solid line shows the results for the Cameraman image with a 30 dB PSNR. The dotted line shows the Cameraman image with a 22 dB PSNR. The dashed line shows the Lenna image with a 30 dB PSNR, and the dash-dotted line shows the Lenna image with a 22 dB PSNR.

utilizing only the scales needed. In addition, the multiscale algorithm is able to provide excellent registration accuracy in the presence of scale changes, which is not even addressed in the Brown algorithm.

4.4 Rotation Performance

The four studies mentioned previously in Section 4.1 will be considered here separately. First, the results will be shown for the comparison study between the scale two rotation registration of the multiscale transform and a single iteration of the algorithm developed by Brown. Next, the results for rotation registration at scales two through six using the new multiscale transform will be presented. With this study, we will also compare the registration accuracy of scale four of the multiscale transform with the second iteration of Brown's algorithm. Then, results will be shown for the rotation algorithm described in Section 3.6 with the reference and input image at the same scale. Finally, the results for the algorithm described in Section 3.6 will be shown for the reference image and input image at different scales. These studies will be separated into two categories. The first three studies fall into

the category of same scale registration and will be considered separately from the last study, which involves registering images at different scales.

4.4.1 Same Scale Rotation Registration Performance. In this section we will discuss registration of images where the input image and the reference image are at the same scale. Three studies were performed in this area. The first study involves comparing the registration of the multiscale transform at scale two with the single scale algorithm developed by Brown [5]. The second study involves looking at multiple scales of the two algorithms. First, we will look at registration accuracy at scales two through six of the multiscale transform. Second, we will register images using the second iteration of Brown’s algorithm and compare those results to equivalent scale four registration of the multiscale transform. Finally, we will show results for the rotation registration algorithm described in Section 3.6.

4.4.1.1 Single Scale Rotation Performance Comparison. The first study was used to determine if the results using a single scale of the multiscale algorithm performed similarly to the Brown algorithm [5]. The results are shown for the rotation algorithms as the number of significant coefficients kept versus the percentage correct. Figure 4.13 shows the results of this study. Figure 4.13(a) shows the results for the Cameraman image with a 30 dB PSNR. Figure 4.13(b) shows the results for the Lenna image with a 30 dB PSNR. Figure 4.13(c) shows the results for the Cameraman image with a 22 dB PSNR, and Figure 4.13(d) shows the results for the Lenna image with a 22 dB PSNR.

For the 30 dB PSNR cases, shown in Figures 4.13(a) and (b), the results for the two algorithms are similar. The multiscale algorithm has some slight deviations from 100% accuracy when registering the Lenna image. Overall, both algorithms have approximately 100% accuracy for the 30 dB PSNR test images. For the 22 dB PSNR cases, Figures 4.13(c) and (d), the multiscale transform has some problems with rotation registration. For the Cameraman image, the multiscale transform

registration performs well, with registration accuracy above 90% for all coefficient values. However, the accuracy of registration for the Lenna image is very poor. Although, the multiscale transform only missed the correct rotation by less than five degrees for the Lenna image. One of the main reasons for this study is to demonstrate that our algorithm at a single scale rivals the results of Brown's algorithm. In the case of the Lenna image with a 22 dB PSNR, we cannot make that claim. This is not necessarily a problem, since our algorithm is a multiscale algorithm. It is most important that the registration accuracy improves when we increase the scale, so that a good accuracy can be obtained using the multiscale algorithm described in Section 3.6.

4.4.1.2 Multiple Scale Rotation Performance Comparison. Now we will look at multiple scale results. The following study shows the registration accuracy of the multiscale transform at scales two through six. Scale four registration using the multiscale transform will be compared to scale two of the Brown algorithm as well.

Figure 4.14 shows the registration performance of scales two through six of the multiscale transform for each of the test images. Since the results for the 30 dB PSNR cases for scale two are so good, we can merely state that in Figures 4.14(a) and (b) the accuracy is good for all scales, and an improvement cannot be shown because the accuracy cannot improve over 100%. Therefore, at all scales, we get the highest accuracy possible. The improvement of registration with increasing scale can more easily be seen in the 22 dB PSNR test cases (Figures 4.14(c) and (d)). For the Cameraman image, the registration is still accurate at scale two, but there are several deviations from 100% accuracy. As the scale increases, there are fewer deviations from 100%, with scale six having no deviations from 100% accuracy. Therefore, we conclude that performance is improving with increasing scale. It is much more obvious in the Lenna image that accuracy is improving with increasing scale. At scale two, the registration accuracy only exceeds 90% in one instance. However, at

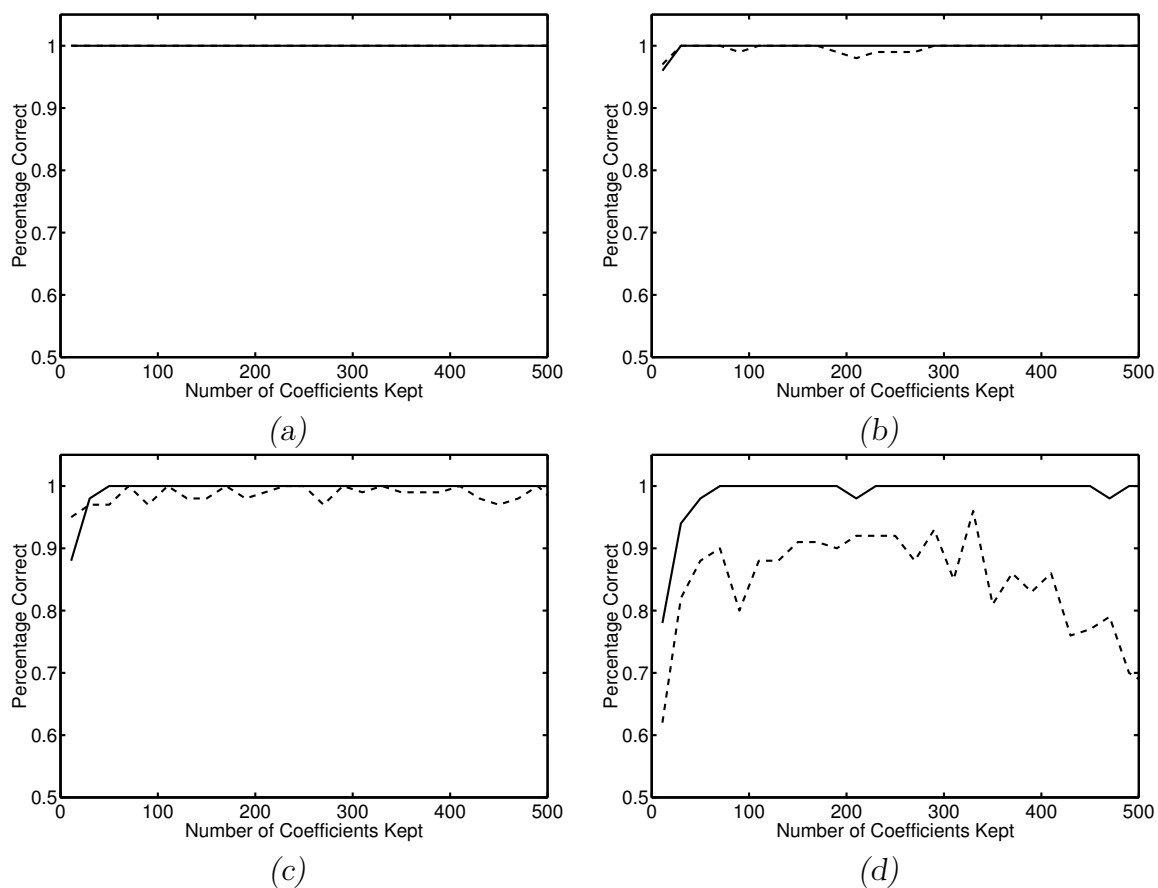


Figure 4.13. Results for scale two rotation registration using the multiscale transform compared to the results of a single iteration of the algorithm developed by [5]. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR. In (a)-(d), the solid line shows the results for the algorithm developed by Brown, and the dashed line shows the results for scale two registration using the multiscale transform.

scale three, the registration accuracy never falls below 90%, and as the scale increases further, the accuracy improves. This trend is especially prevalent in the region where the number of coefficients retained is small. Clearly, we should apply a multiscale algorithm that utilizes as many scales as possible to achieve the correct registration.

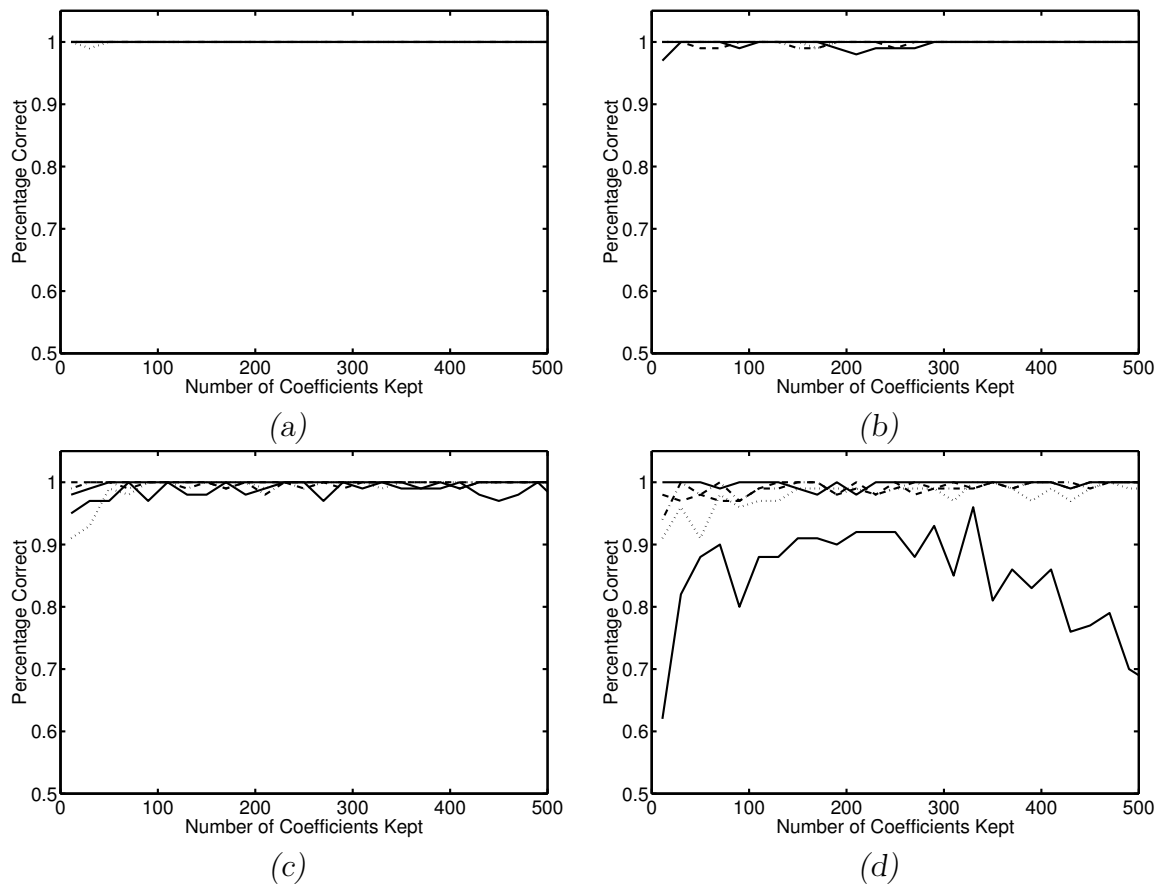


Figure 4.14. *Results for rotation registration using scales two through six of the multiscale transform. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR. In (a)-(d), the lower solid line shows scale two of the multiscale transform, the dotted line shows the scale three results, the dashed line shows the scale four results, the dot-dashed line shows the scale five results, and the scale six results are shown by a solid line as well. The scale six results are typically around 100%.*

Figure 4.15 shows the comparison of scale four registration accuracy of the multiscale transform and the second iteration registration of the Brown algorithm. Registration using the multiscale transform is essentially the same as the registra-

tion using the second iteration of Brown’s algorithm. For the 30 dB PSNR cases, Figures 4.15(a) and (b), the registration in both cases is essentially 100%. However, there are a few cases in the Lenna image where the registration using the multiscale transform falls slightly below 100%. Virtually the same observation can be made about registration of the images with a 22 dB PSNR, Figures 4.15(c) and (d). In the Cameraman case, the accuracy of registration using the multiscale transform is slightly better than the Brown algorithm at lower numbers of coefficients kept. There are a few deviations from 100% accuracy in the higher numbers of coefficients when using the multiscale transform to register the image that are not present in the algorithm by Brown. However, overall, the performance of both algorithms is essentially the same. For the Lenna image, the registration accuracy at small numbers of coefficients is also better than the Brown algorithm. However, the Brown algorithm registration accuracy quickly goes to 100%, whereas the registration accuracy of the multiscale algorithm just lingers around 100%. We recognize that the multiscale transform registration is slightly worse at scale four than the second iteration of the Brown algorithm, but overall, the results for both algorithms are very accurate.

4.4.1.3 Rotation Performance for Same Scale Test Images. The purpose of this study is to determine the accuracy of the algorithm described in Section 3.6 for the case where the input image and the reference image are at the same scale. The minimum Pearson correlation value used for this algorithm was determined to be 0.96 in Section 3.7.6. To improve accuracy, the multiscale algorithm must spend more time to determine the rotations that are needed because a greater scale must be used. The multiscale algorithm that we have developed seeks to find an adequate trade-off where we can obtain the most accuracy with the least amount of time spent. Therefore, two parts are needed in this study. The first aspect of this study is to determine if this algorithm provides accurate registration, and the second aspect is to determine if the algorithm will work quickly. To demonstrate accuracy, we present the registration results using this algorithm in Figure 4.16.

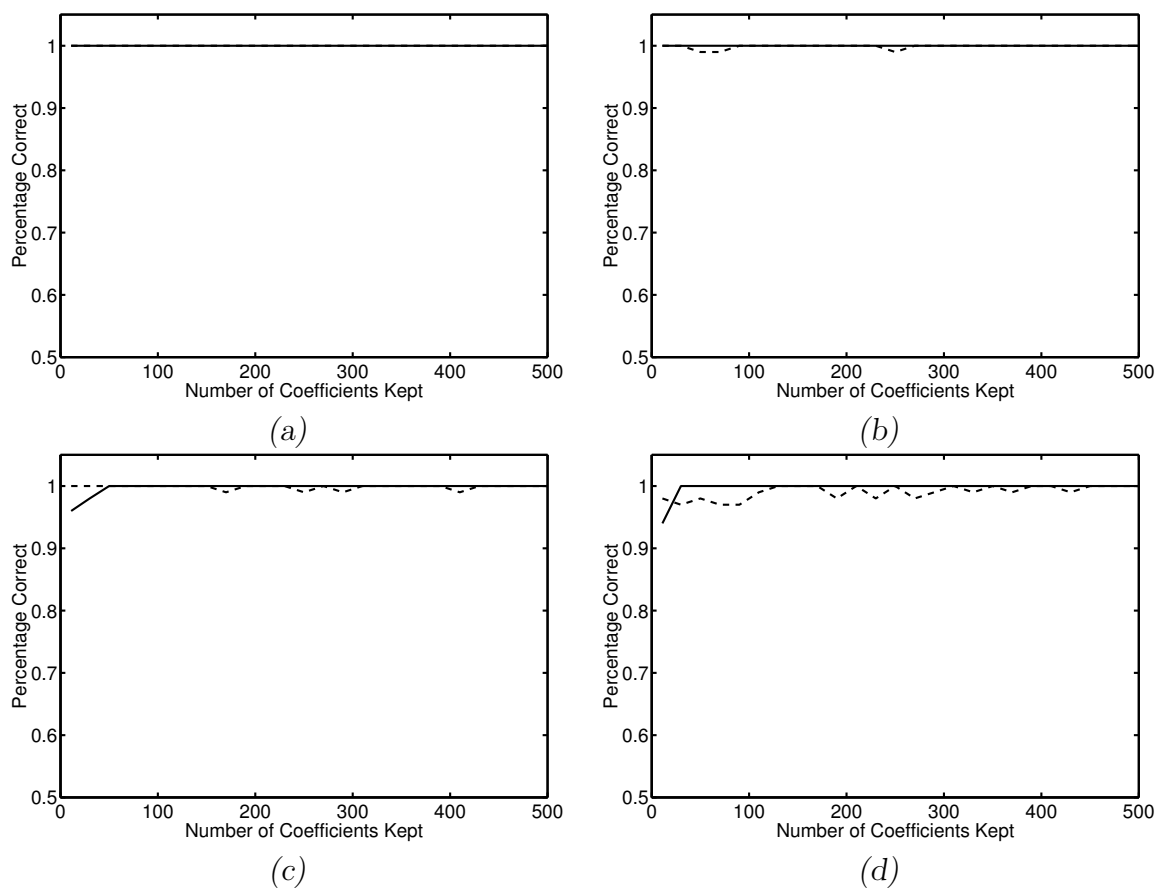


Figure 4.15. Results for scale four rotation registration using the multiscale transform compared to the results of the second iteration of the algorithm developed by Brown [5]. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR. In (a)-(d), the solid line shows the results using the algorithm developed by Brown, and the dashed line shows the results for scale four registration using the multiscale transform.

This algorithm performed extremely well. In all cases, the accuracy was 100%. In determining rotation registration, the 0.96 minimum Pearson correlation value is an adequate choice.

Any number of coefficients could be kept. However, since the accuracy of the registration algorithm seems to be somewhat image dependent, a small number of coefficients may not be a good choice. To increase the robustness of the algorithm, a value between 100 and 250 coefficients would work well for registration. If we require the algorithm to run quickly, we would choose a coefficient value at the low end of this range, and if we want possibly better accuracy, we would choose a higher coefficient value. The choice of coefficients retained permits the trade-off between time and accuracy.

The second aspect of this study is to determine if our multiscale algorithm can produce accurate results in a short amount of time by utilizing only the scales needed to satisfy the minimum Pearson correlation value. Once again this is difficult to show by simply showing computation time due to the varying speeds of the computers used to compute the registration results. Since we cannot use computational time, we will show that our algorithm uses the smallest scale necessary to produce accurate registration, thus it is efficient and will register images quickly. We can also make comparisons with the computational time of the single scale Brown algorithm.

Brown's algorithm [5] is iterative. Therefore, in order to create the appropriate subbands at a larger scale, we must first create the subbands at the scale below it. If we choose a scale above scale one, additional time would be spent computing the transform to obtain the appropriate coefficients. The multiscale algorithm obtains registration estimates at each successive scale until the necessary minimum Pearson correlation is met. In both cases, the greater the scale, the more time the algorithm takes to run. In order for the Brown algorithm to rival the accuracy of our multiscale algorithm, it would have to use a scale that is equal to or larger than the average scale used to register the image in the multiscale algorithm. The multiscale algorithm

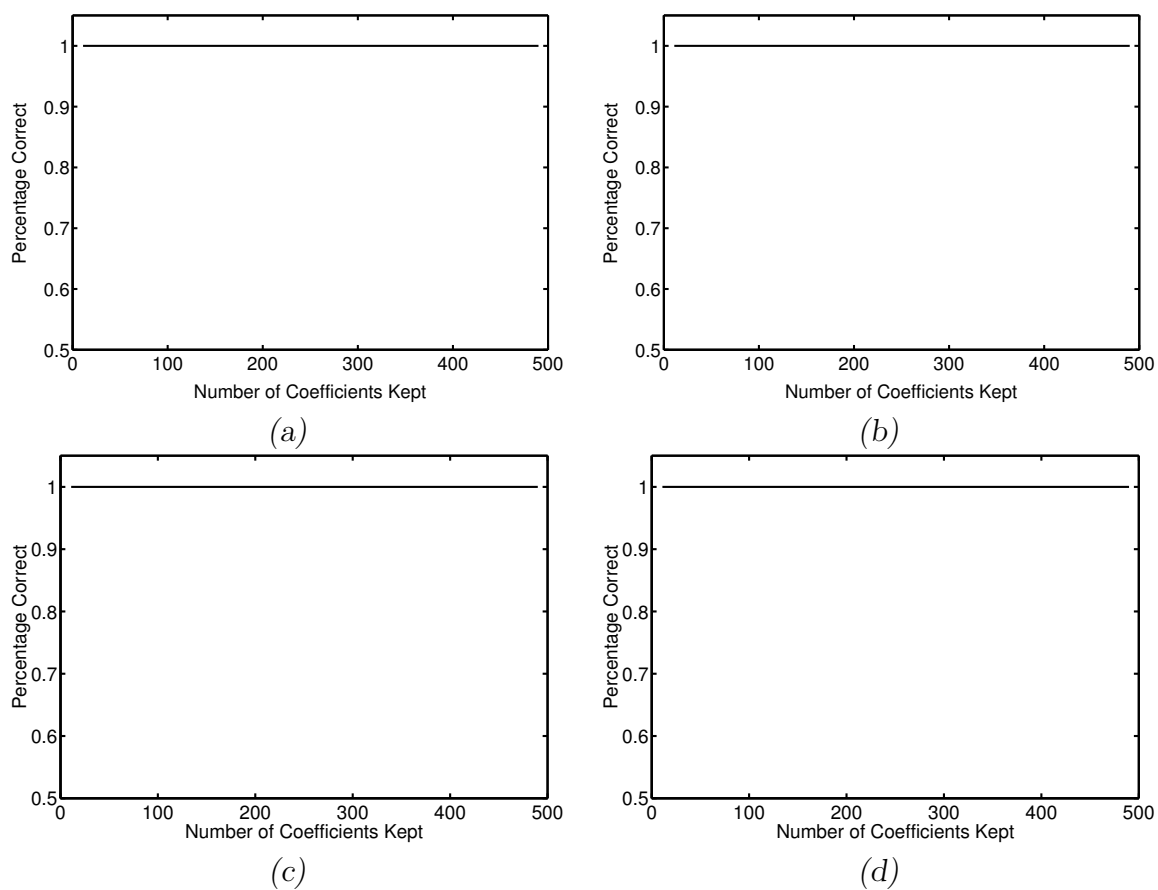


Figure 4.16. *Results for the multiscale registration algorithm for rotation registration for images at the same scale. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR.*

should perform better than Brown’s algorithm at the appropriate scale, since some images will require a scale greater than this single scale to register correctly. Our multiscale algorithm allows us to utilize the larger scales if needed, while keeping the computational time down by only using the scale necessary to achieve a desired minimum Pearson correlation. Therefore, the average time needed to compute the correct registration estimate using the multiscale algorithm should be about the same overall compared to the Brown algorithm at the appropriate scale, while the accuracy of our multiscale algorithm should be better. To determine the computational time of our algorithm, the average scale used to register the images for each number of significant coefficients was calculated. The results for this study are shown in Figure 4.17.

In the 30 dB PSNR cases, the average scale used for registration was scale two. Therefore, in this instance the Brown algorithm would probably perform quicker with the same amount of accuracy. However, since we are looking for a robust transform capable of registering several different types of images, the Brown algorithm may still not be the best choice. Images with higher amounts of noise require larger scales to register correctly, and the single scale Brown algorithm cannot account for this.

For the 22 dB PSNR cases, all scales were utilized without achieving the required minimum Pearson correlation, so the best Pearson correlation was chosen. If the minimum Pearson correlation value was reduced, the scale used would probably be less with approximately the same amount of accuracy. This would allow the algorithm to stop processing at lower scales, and the algorithm would run faster.

4.4.2 Rotation Performance for Test Images at Different Scales. The purpose of this study was to determine the accuracy of the algorithm described in Section 3.6, where the input is at a different scale compared to the reference image. The same method as described in Section 4.4.1.3 is used. A minimum Pearson correlation value of 0.96 is used in this study. The first part of this study shows the algorithm’s

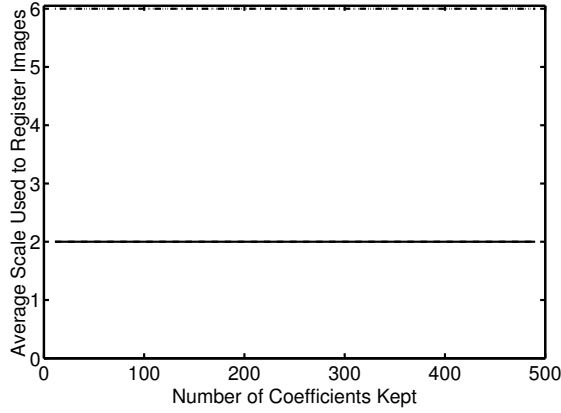


Figure 4.17. *Average scale used for each test image to register the input image with rotation differences to the reference image when the images are at the same scale. The solid line shows the results for the Cameraman image with a 30 dB PSNR. The dotted line shows the Cameraman image with a 22 dB PSNR. The dashed line shows the Lenna image with a 30 dB PSNR, and the dash-dotted line shows the Lenna image with a 22 dB PSNR.*

accuracy for different numbers of significant coefficients. These results are shown in Figure 4.18. Once again, the registration accuracy is essentially 100% in all test cases. Therefore, we can once again say that 0.96 seems to be an adequate choice for the minimum Pearson correlation value where rotation registration is concerned.

To recommend an appropriate value of coefficients to retain, more research needs to be done on additional images, since the registration accuracy seems to be somewhat image dependent. We hypothesize that a good range is between 100 and 250 coefficients, so that the algorithm is robust. In this range, our algorithm is robust and efficient, while the effects of the noise are relatively small.

The second part of this study is to show that the multiscale algorithm performs quickly, with a greater accuracy than a single scale algorithm. We have already shown that the accuracy is greater than or equal to the accuracy of a single scale algorithm. Now, we must show that its speed is comparable. The results for this study are shown in Figure 4.19. For the 30 dB PSNR case, we would tend to need a single scale algorithm at scale three to obtain a similar accuracy to the multiscale

algorithm. For the Brown algorithm, we are forced to use scale two. Since some of the images need a scale greater than two to register correctly, the accuracy of the Brown algorithm would be less than the accuracy of the multiscale algorithm. Also, the time would tend to be about the same because the multiscale algorithm only uses the scale necessary to determine the appropriate accuracy. In some cases, this would only require the use of scale two. In other cases, scales up to scale six would be necessary. The average scale used for the 30 dB noise cases is between scales two and three, so this algorithm would take about the same amount of time as a single scale algorithm at scale three. We require an algorithm that performs well in all cases, and therefore for the 22 dB PSNR case we would typically utilize scales near six. To equal the performance of our algorithm, a scale six single scale algorithm would need to be used. In this case, our algorithm should perform faster because our algorithm only utilizes the scale necessary to achieve the given minimum Pearson correlation value. Thus, our multiscale transform provides improved accuracy and increased flexibility with little or no additional processing requirements.

4.5 *Summary*

In most cases, the single scales of the multiscale algorithm perform equally to the algorithm developed by Brown. The multiscale algorithms described in Sections 3.5 and 3.6, performed well for the noise cases that we used. The 0.96 minimum Pearson correlation value seems to be an adequate value to produce good image registration accuracy. If some information is known about the noise in the images, then the minimum Pearson correlation value can be adjusted to allow the program to use fewer scales to produce the same registration accuracy. The general rule is that the minimum Pearson correlation value can be lowered for higher noise cases. This is because with larger amounts of noise, there is less correlation between two registered images. These studies show that the multiscale registration algorithm has better accuracy than the single scale algorithm developed by Brown [5], and therefore would

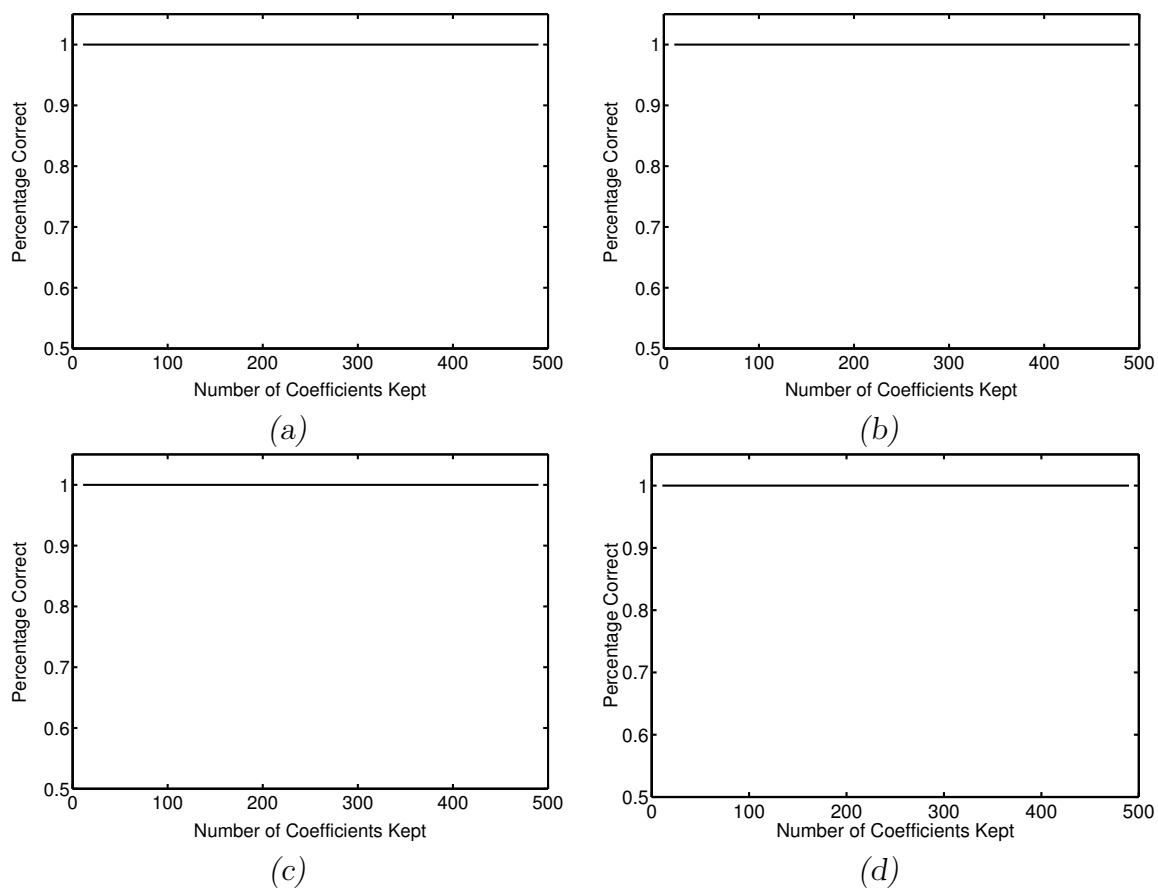


Figure 4.18. *Results for the multiscale registration algorithm for rotation registration where the input image and reference image are at different scales. (a) Cameraman with a 30 dB PSNR. (b) Lenna with a 30 dB PSNR. (c) Cameraman with a 22 dB PSNR. (d) Lenna with a 22 dB PSNR.*

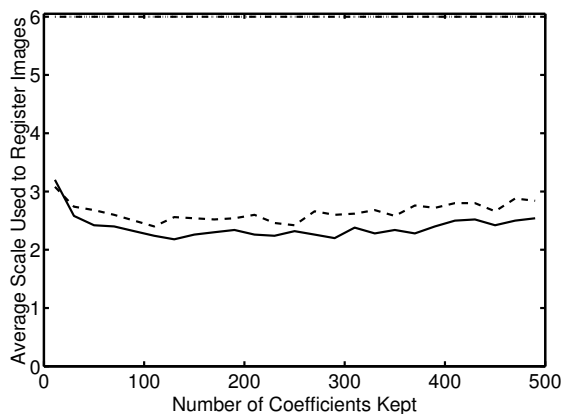


Figure 4.19. Average scale used for each test image to register the input image with a rotation difference to a reference image when the images are at different scales. The solid line shows the results for the Cameraman image with a 30 dB PSNR. The dotted line shows the Cameraman image with a 22 dB PSNR. The dashed line shows the Lenna image with a 30 dB PSNR, and the dash-dotted line shows the Lenna image with a 22 dB PSNR.

be preferred over this algorithm. Our multiscale algorithm has improved versatility over a single scale algorithm due to the option of computing multiple scales. Also, these advantages come at very little cost, since the minimum Pearson correlation requirement allows the algorithm to finish without computing the registration estimates at all scales.

V. Discussion and Future Work

5.1 Contributions of this Thesis

There are several new ideas for image registration developed in this thesis. First, improvements were made to the masking process. We incorporated an exclusion zone to enforce spatial separation of features. These improvements force more spacing between significant coefficients, which improves registration performance. This masking technique was implemented for both translation and rotation registration.

Secondly, additional scales of the algorithm developed by Brown [5] were created and utilized in these studies. A new multiscale transform was developed which utilizes the continuous wavelet transform and a lowpass filter to mimic the redundant discrete wavelet transform. This new transform was then used in a registration algorithm to adequately predict translations and rotations of an input image compared to a reference image in the presence of noise and scale differences.

Finally, a method was created to utilize only the scales needed to obtain the desired accuracy, using the minimum Pearson correlation. An adequate minimum Pearson correlation value was determined for this algorithm if the amount of noise in the images is unknown. The general rule is that a higher noise value requires a lower minimum Pearson correlation value to provide accurate registration.

The purpose of switching the algorithm from a redundant discrete wavelet transform algorithm to a multiscale continuous wavelet transform/lowpass filter algorithm is that our algorithm allows the utilization of multiple scales to produce accurate registration. Also, a scale-invariant transform using the continuous wavelet transform can potentially be developed. In this case, a single transform could be used to determine all of the changes needed to make the two images match.

In this research, we have developed a multiscale registration algorithm. Within this algorithm, we have developed a new multiscale transform and a way of utilizing the scales of this transform to provide accurate registration. We have also developed a new method of masking with the implementation of exclusion zones which provides spatial separation of the significant coefficients. From the results of this study, we conclude that this algorithm has excellent registration performance with respect to translation or rotation differences, independent of scale changes and additive noise. This algorithm out performs single scale algorithms with respect to accuracy, and this accuracy comes at a low cost, because we have minimized the time required to complete registration by utilizing only the scales necessary to provide accurate registration.

5.2 *Potential for Future Research*

Several new topics of interest have been explored in this research. These topics present new ideas for future theses.

5.2.1 *Implementation of a Fast Continuous Wavelet Transform Algorithm.*

The continuous wavelet transform must be computed by approximating integrals. This is a very computationally intensive process and is therefore slow. In order to make this algorithm more efficient, a fast continuous wavelet transform that can be computed on integer scales needs to be implemented. Several algorithms have been developed to create a fast implementation of the continuous wavelet transform on integer scales [3, 22, 29, 37, 38]. One of these algorithms could potentially be utilized to increase the speed of our multiscale algorithm. In fact, we designed our algorithm around integer scales to facilitate the incorporation of a fast transform. Thus, this fast continuous wavelet transform implementation would simply be substituted for the continuous wavelet transform algorithm that is currently used with little adjustment to the multiscale registration algorithm.

5.2.2 *Determining Appropriate Values for the Minimum Pearson Correlation.*

We determined an adequate minimum Pearson correlation value for translation registration. Experimentally, we found that this minimum Pearson correlation value also accurately registered images with rotation differences as well as scale changes. More experimentation needs to be done to determine if the minimum Pearson correlation value that we chose is appropriate for registration of rotation differences and scale changes, or if there is a better value to increase the efficiency of the algorithm.

Additionally, if information is known about the amount of noise in an image, this minimum Pearson correlation value can be adjusted to provide accurate results more quickly. Adjustments could be made adaptively depending on the amount of noise in a particular image or a better minimum Pearson correlation value for the set of images could be determined if it is known that all images had similar amounts of noise.

5.2.3 *Creating an Algorithm to Register Scales.* The differences between two images typically occur in three categories: translations, rotations, and scale changes. Translation and rotation registration algorithms have already been developed, and our multiscale algorithm registers images at different scales. However, it would be helpful to estimate the scale differences between two images. Some attempts were made to create a scale-invariant transform in this thesis. Although this algorithm has potential, the results are sporadic, and more work needs to be done in this area.

In order to adequately register images, we must create an algorithm that will find the scale changes, and rescale one of the images, so that both images are the same size. This is a difficult problem, since large coefficients persist across scale. The development of the fast continuous wavelet transform algorithm will aid in the efficiency of this method. With the scale-invariant registration algorithm, the fast

continuous wavelet transform method would need to be extended to non-integer scales, so that we would be able to home in on the scale change.

5.2.4 Calculate Translation, Rotation, and Scale Changes Simultaneously.

Registration of strongly translated or rotated images can be accomplished using algorithms developed in this thesis. Once the scale-invariant algorithm is developed, scale changes will be able to be registered as well. However, all three of these changes can be present in a single image when compared to a reference image. Therefore, it is necessary to create an algorithm that can determine the translations, rotations, and scale changes between two images simultaneously. One other feat associated with this idea is to determine which ordering of these operations is required.

5.2.5 Creating a Super Resolution Algorithm. The ultimate goal of this problem would be to use the registration information in a super resolution algorithm. The translations, rotations, and scale changes would be determined, and the images would be shifted to the same plane using these values. Once this is complete, the images can be placed on top of each other to form a “super” image, where hopefully more information could be gained than from any of the individual images by themselves.

Bibliography

1. Allen, R. L., et al. "Laplacian and Orthogonal Wavelet Pyramid Decompositions in Coarse-to-Fine Registration," *IEEE Transactions on Signal Processing*, 41(12):3536–3541 (December 1993).
2. Alliney, S. and C. Morandi. "Digital Image Registration Using Projections," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(2):222–233 (1986).
3. Berkner, K. and Jr. R. O. Wells. "A Fast Approximation to the Continuous Wavelet Transform with Applications." *IEEE Proceedings of the 31st Asilomar Conference on Signals, Systems, and Computers*. 1997.
4. Brown, L. G. "A Survey of image Registration Techniques," *ACM Computing Surveys*, 24(4):325–376 (December 1992).
5. Brown, R. K. *Image Registration using Redundant Wavelet Transforms*. MS thesis, Air Force Institute of Technology, Wright-Patterson AFB, OH, March 2001.
6. Burrus, C. S., et al. *Introduction to Wavelets and Wavelet Transforms: A Primer*. Englewood Cliffs, NJ: Prentice-Hall, 1997.
7. Castro, E. De and C. Morandi. "Registration of Translated and Rotated Images Using Finite Fourier Transforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):700–703 (1987).
8. Claypoole, R. L. *Multipoint Multirate Signal Processing*. MS thesis, Air Force Institute of Technology, Wright-Patterson AFB, OH, December 1994.
9. Corvi, M. and G. Nicchiotti. "Multiresolution Image Registration." *Proceedings of IEEE International Conference on Image Processing*. 224–227. October 23-26 1995.
10. Crouse, M. S., et al. "Wavelet-Based Statistical Signal Processing Using Hidden Markov Models," *IEEE Trans. Signal Processing*, 46:pp. 886–902 (1998).
11. Daubechies, I. *Ten Lectures on Wavelets*. CBMS-NSF Regional Conf. Series in Appl. Math., Vol. 61, Philadelphia, PA: SIAM, 1992.
12. Djamjdi, J. P., et al. "Geometrical Registration of Images. The Multiresolution Approach," *Photogrammetric Engineering and Remote Sensing Journal*, 59(5) (May 1993).
13. Edwards, C. H. and D. E. Penney. *Calculus and Analytic Geometry*. Englewood Cliffs, New Jersey: Prentice Hall, 1990.

14. Kaiser, G. *A Friendly Guide to Wavelets*. Boston, MA: Birkhauser, 1994.
15. Kuglin, C. D. and D. C. Hines. "The Phase Correlation Image Alignment Method." *Proceedings of IEEE International Conference on Cybernetics and Society*. 163–165. September 1975.
16. LeMoigne, J. "Parallel Registration of Multi-Sensor Remotely Sensed Imagery Using Wavelet Coefficients." *OE/Aerospace Sensing, Wavelet Applications Conference, 2242*. 432–443. April 5-8 1994.
17. LeMoigne, J. "Towards a Parallel Registration of Multiple Resolution Remote Sensing Data." *International Geoscience and Remote Sensing Symposium*. 1011–1013. July 10-14 1995.
18. LeMoigne, J., et al., "An Automated Parallel Image Registration Technique of Multiple Source Remote Sensing Data." *CESDIS TR-96-182* - submitted to IEEE Transactions in Geoscience and Remote Sensing.
19. LeMoigne, J. and R. F. Crompt. "The Use of Wavelets for Remote Sensing Image Registration and Fusion." *Wavelet Applications, 2762*. 535–544. 1996.
20. LeMoigne, J. and I. Zavorin. "An Application of Rotation- and Translation-Invariant Overcomplete Wavelets to the Registration of Remotely Sensed Imagery." *SPIE's OE/Aerospace Sensing, Wavelet Applications Conference*. 130–140. April 6-8 1999.
21. LeMoigne, J. and I. Zavorin. "Use of Wavelets for Image Registration." *Wavelet Applications VII 4056*, edited by H. Szu, et al. 99–108. 2000.
22. Lewis, J. M. and C.S. Burrus. "Approximate Continuous Wavelet Transform with an Application to Noise Reduction." *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing III*. 1533–1536. May 12-15 1998.
23. Li, H. H. and Y. Zhou. "A Wavelet-Based Feature Extractor for Multi-Sensor Image Registration." *Proceedings of the SPIE, 2762*. 524–534. 1996.
24. Lim, J. S. *Two-Dimensional Signal and Image Processing*. New Jersey: Prentice Hall, 1990.
25. Mallat, S. G. "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693 (1989).
26. Manjunath, B. S., et al. "A New Approach to Image Feature Detection with Applications," *Elsevier Science Pattern Recognition*, 29(4):627–640 (1996).
27. Marr, D. *Vision*. New York: W. H. Freeman, 1982.
28. Meng, T. H., "Class Handout EE265: Digital Signal Processing, Lecture 5: FIR Filter Design by Windowing." Department

- of Electrical Engineering, Stanford University, Stanford CA, 2001.
<http://dualist.stanford.edu/ee265/handouts/lecture5.pdf>.
29. Munoz, A., et al. "Fast Continuous Wavelet Transform Based on B-Splines." *Proceedings of the SPIE Conference on Mathematical Imaging: Wavelet Applications in Signal and Image Processing IX* 4478. July 29-August 1 2001.
 30. Reddy, B. S. and B. N. Chatterji. "An FFT-Based Technique for Translation, Rotation, and Scale-Invariant Image Registration," *IEEE Transactions on Signal Processing*, 3(8):1266–1270 (August 1996).
 31. Sharman, R., et al. "Compensating for Wavelet Sensitivity to Translation in Image Registration Applications." *Wavelet Applications VI*, 3723. 466–468. 1999.
 32. Sharman, R., et al. "A Fast and Accurate Way to Register Medical Images Using Wavelet Modulus Maxima," *Elsevier Science Pattern Recognition Letters*, 21:447–462 (2000).
 33. Stone, H. S., et al. "The Translation Sensitivity of Wavelet-Based Registration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):1074–1081 (October 1999).
 34. Strang, G. and T. Nguyen. *Wavelets and Filter Banks*. Wellesley, Cambridge, 1996.
 35. Tashakkori, R., et al. "Prediction of Medical Images Using Wavelets." *Wavelet Applications VII*, 4056, edited by H. Szu, et al. 332–340. 2000.
 36. The MathWorks, Inc., "Matlab Release 12.1 Documentation, Signal Processing Toolbox, Filter Design, FIR Filter Design." <http://www.mathworks.com/access/helpdesk/help/toolbox/signal/signal.shtml>, 2001.
 37. Unser, M., et al. "Fast Implementation of the Continuous Wavelet Transform with Integer Scales," *IEEE Transactions on Signal Processing*, 42(12):3519–3522 (December 1994).
 38. Vrhel, M. J., et al. "Rapid Computation of the Continuous Wavelet Transform by Oblique Projections," *IEEE Transactions on Signal Processing*, 45(4):891–900 (April 1997).

Vita

Jennifer L. Manfra graduated from the University of Memphis in 1999 with a B.S. in Electrical Engineering. She was selected as a direct assession to the Air Force Institute of Technology (AFIT) at Wright-Patterson AFB, OH where she received her M.S. in Electrical Engineering in March of 2002. Lieutenant Manfra's next assignment will be to the Air Force Research Laboratory at Wright-Patterson Air Force Base, where she will be working in the Sensors Directorate with the Find, Fix, Track, and ID Branch.

Permanent address: AFIT/ENG, 2950 P St.
Wright-Patterson AFB, OH 45433-7765

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 074-0188	
<p>The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY) 15-03-2002		2. REPORT TYPE Master's Thesis		3. DATES COVERED (From – To) Jun 2001 – Mar 2002	
4. TITLE AND SUBTITLE Translation and Rotation Invariant Multiscale Image Registration				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Manfra, Jennifer L., 2d Lt, USAF				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 P Street, Building 640 WPAFB OH 45433-7765				8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GE/ENG/02M-16	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFRL/IFED Attn: Maj. John G Keller 32 Brooks Road Rome NY 13441-4114 DSN: 587-3944 e-mail: john.keller@rl.af.mil				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.					
13. SUPPLEMENTARY NOTES					
<p>14. ABSTRACT With recent technological advances, image data is being collected in larger quantities. This data must be analyzed in an accurate and efficient manner. Human analysis of data is time consuming and inaccurate, so a fast, automated, accurate method to register images is needed. Image registration is a key step in analyzing imagery data for super resolution and allows easy comparison of images. Recent research registered images in the presence of translations and rotations using one iteration of the redundant discrete wavelet transform (DWT). We extend this by creating a new multiscale transform to register images with translation or rotation differences, independent of scale differences. Our two-dimensional multiscale transform uses a lowpass filter and the continuous wavelet transform to mimic the two-dimensional redundant DWT to obtain multiple subbands at various scales while maintaining the desirable properties of the DWT. The DWT produces results at dyadic scales; our multiscale transform produces data at integer scales. This added flexibility improves registration accuracy without greatly increasing computational complexity and permits accurate registration in the presence of scale differences. We demonstrate the performance of our algorithm by registering images at various rotations and translations, in the presence of scale differences and additive white noise.</p>					
15. SUBJECT TERMS Image Registration, Continuous Wavelet Transform, Multiscale Transform					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Roger L. Claypoole, Maj, USAF (ENG)
U	U	U	UU	113	19b. TELEPHONE NUMBER (Include area code) (937) 255-3636, ext 4625; e-mail: roger.claypoole@afit.edu